

# Fido

## The Book

Preliminary Documentation to Fido V12

August, 1987



h - 415-882-9832  
or

Fido Software  
Tom Jennings  
164 Shipley St.  
San Francisco CA 94107

voice: (415)-764-1688  
data: (415)-764-1629  
fido: 1:125/111 Fido Software





# Fido

# The Book

Preliminary Documentation to Fido V12

August, 1987

Written by Scot Kamins

With assistance from Tom Jennings and Tim Pozar

Fido Software  
Tom Jennings  
164 Shipley St.  
San Francisco CA 94107  
(415)-764-1688

Fido™ is a trademark of Tom Jennings

FidoNet® is a registered trademark of Tom Jennings

The dog-with-diskette drawing is a trademark of Tom Jennings

This manual and it's contents copyright Tom Jennings August 1987. None of this material may be copied in any form without prior written permission



# Acknowledgements

The following people have contributed greatly to the design of Fido and FidoNet, starting in 1984. While it was long enough ago that the particular features they contributed to are now "obvious", at the time they were revolutionary, and form the backbone of what constitutes "IFNA" and the public access FidoNet network. There are many names missing from this list, and I apologize for leaving their names out. These people in particular contributed more than anyone could ever expect. And special thanks to John Madill, who thought out many of the nicer Fido features, very early in Fido's life. (He finally got me to do PickUp and Poll ...)

## Fido/FidoNet design & debug

John Madill, Baltimore MD	Fido #2
Tony Clark, St. Louis MO	Fido #4
Danny Feinsmith, NYC, NY	Fido #8
Jim Ryan, Danbury CT	Fido #9
Ben Baker, St. Louis MO	Fido #10
Vern Crawford, San Jose CA	Fido #13

## FidoNet scheduling and routing

The St. Louis DECUS Group:	
Ken Kaplan	Fido #22
Mike Mellinger	Fido #16
Jon Wichman	Fido #17

In more recent years, with the IFNA net getting large enough to require a new way of thinking, the following people spring to mind when I think of who helped tie the net together into a coherent technical standard:

Randy Bush	1:105/6
Bob Hartman	1:132/101
Ken Kaplan	1:100/22
Thom Henderson	1:107/1
Gee Wong	1:107/312

And saving the best and most important for last, so you will remember it; the person who single-handedly organized the network and infrastructure that became the IFNA net and the IFNA corporation; who ran the national network host since 1984; who contributed more technical, organizational and just plain help than any person or group of people involved since the start; who was level headed when the net went crazy.

Ken Kaplan	1:100/22
------------	----------

Ken also at the same time devoted more time and energy to bulletin board users and their legal rights to use them than anyone else I can think of. Anyone who has even used a bulletin board in the last few years owes Ken thanks.

And thanks also to Sally Kaplan, for putting up with us all. Their two kids have pretty cool parents, too.

## A Special Acknowledement from the author

The author would like to thank Tim Pozar, System Operator of KKSF BBS, San Francisco, for his technical help in preparing this manual. He read this manual more times than anybody should have to, and caught more technical errors than the author would ever admit to. Of course, any remaining technical errors are the responsibility of the author (although we all know that any such errors weren't corrected because Pozar missed them.)



## About This Documentation

This is the preliminary documentation for Fido Version 12. It is complete and it is accurate, but it lacks tips and tricks and other good stuff that comes only after Software has been used for a while.

It costs us about \$7.00 to make and mail a manual. If you'll tear this page out and mail it to us, along with your feedback about this manual and \$7.00 (payable on a United States bank), we'll mail you new documentation as soon as it's available.

You must tear out this sheet; you cannot send a photocopy.

Name: \_\_\_\_\_

Address: \_\_\_\_\_

City \_\_\_\_\_

State/Province/District \_\_\_\_\_

Country/Zip/Postal Zone \_\_\_\_\_

Typos, inaccuracies, complaints, compliments, things we should have included:

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

(Write more on the other side or attach a separate sheet. We'll pay attention to what you say. Honest!)

This is the ...  
...  
...

...  
...  
...

You ...

...

...

...

...

...

...

...

...

...

...

...

...

...



## Table of Contents

Introduction	1
What You Need to Run Fido	1
Compatibility Issues	2
Fido and Disk Space	2
What You Must Already Know	2
What's In This Manual	3
A Plea For Feedback	3
 <b>Chapter 1- Setting Up Fido</b>	 4
First Steps	5
Customizing Fido's Files	6
The .INI Files- Fido's Start-up Material	6
FIDO.INI	7
Table of Preset FIDO.INI variables	8
Time and Download Limits	9
DOS ERRORLEVEL Values	9
Modem Information	10
FidoNet Parameters	10
Miscellaneous Parameters	12
AREAS.INI	13
Defining an Area	13
COMMANDS.INI	16
Changing Command Names	16
Setting Access Levels	18
About Privileges	18
About Locks	18
Table of Preset Command Names and Privileges	19
EVENTS.INI	21
Event Line Components	21
The Special OPT Keyword	22
Event Execution	23
Customizing System Notices	24
How Help Works	26
The Questionnaire	27
Constructing the Questionnaire	27
Reading the Answer File	28
Three Kinds of Questionnaire	28
The Private System's Screening Questionnaire	29
The Public or Semi-Private System's Questionnaire	29
Setting Up a Private System	30
Display a File of Information	30
New Caller Questionnaire	30
InfoFile in Questionnaire Disguise	30
Semi-Private Systems	31
Big Bang Time	32
Nothing Can Go Wrong (scratch) Go Wrong (scratch) Go...	32
 <b>Chapter 2- Running the BBS</b>	 36
The System Operator's Job	36
Console Commands	37
Command Line Switches	39
Managing Uploaded Files	40
Adding Notes	40
Using @ to Limit New File Display	41
Moving @ with the K)ill Command	42
SYSOP.EXE - Fido's Caller Clean-Up Utility	43
Small South-American Country Time	45

MSGMGR.EXE - Automated Cleanup	46
TIMELOG.EXE - Fido's Call Counter	47
FIDO.LOG - Fido's Caller and Event Log	48
<b>Chapter 3- FidoNet</b>	49
Setting Up Your Own Network	49
Choosing a Network Topology	50
Joining the IFNA Network	51
Nodes, Nets, Regions, and Zones	51
How Mail Gets Sent	52
FidoNet Events	52
Route Files- FidoNet's Message Forwarding System	52
Route File Commands	53
Node List	56
What the List Looks Like	57
Compiling the Nodelist	57
<b>Chapter 4- Fido V12 Caller's Guide</b>	58
Fido's Structure	58
Calling Fido	59
About Names	60
First Time Log-Ons	60
Choosing a Password	60
The Main Menu	61
Changing Caller Settings	63
Help Levels	64
Message Section	66
Special Message Commands	71
Sending and Receiving FidoMail	73
Setting Up a FidoNet Account	73
Using the FidoNet Message System	73
The N Command	74
The F Command	75
Sending a File	75
Files Section	77
Fido's Command Buffer	
An Undocumented Feature	81
<b>Appendix A- Delta Guide to Version 12</b>	82
Converting from Version 11 to Version 12	82
Running 11-TO-12.EXE	82
File Name and Format Changes	84
<b>Appendix B- Files and their Functions</b>	85
<b>Appendix C- Modem Basics</b>	87
Modem Initialization	87
Going OnHook (Disconnecting)	89
<b>Index</b>	91



## Introduction

FIDO<sup>®</sup> is a complete stand-alone BBS/electronic packet mail system. Use the bulletin board section as a public or private center for up to 200 special interest topics and as a general medium for the exchange of information; use the mail system to exchange files and messages of any length with any of 2000 other FidoMail nodes in 50 states and 17 countries, or set up a private electronic mail network within your own organization.

## What You Need to Run Fido

To run Fido, you need all of the following:

- MS-DOS version 2.0 or higher
- Any of the following computers with a minimum of 2 disk drives and at least 192K of free memory :
  - IBM PC, XT, AT, PS/2.
  - IBM exact hardware clones
  - DEC Rainbow 100 (running MS-DOS 2.05 or higher)
  - Victor 9000 (with IBM-compatible drives only)
- Any of the following asynchronous modems:
  - DC Hayes Smartmodem 1200 & 2400
  - DC Hayes exact clones
  - Multi Tech 2400
  - Prentice POPCOM
  - US Robotics 212A, Courier 2400, or Courier HST 9600
- A text editor or word processor capable of reading and producing ASCII files (which includes almost all word processors available today)

Fido might run on hardware not listed here, but Fido Software doesn't guarantee it. At the moment, these are the *only* systems that Fido Software supports. There are also many other "Hayes-compatible modems not listed here; some work and some don't; all we can suggest is that you try it and see. Many times a marginally-compatible modem can be coerced into working properly with the right commands; please refer to "Appendix C: Modem Basics" in case you want to try.

FIDO needs 192K of free memory. That's *really* free memory - use CHKDSK to find out how much memory is really available.

## Compatibility Issues

Fido makes extensive use of interrupts. Memory-resident programs like NEWKEY, pop-up alarm clocks, SIDE KICK, and other programs of their ilk interfere with the process by doing strange and evil things with the system. You might not notice any problems at first, but one day when you least expect it ...

Fido is, however, compatible with all versions of PC-DOS starting with Version 2.00, IBM's TOPVIEW, Microsoft Windows, Digital Research's concurrent PC-DOS Versions 4.0 (and later), most LAN's including ViaNet, Novell's NetWare, and many multitaskers like Software Link's Multilink.

## Fido and Disk Space

A typical Fido system (including MS-DOS system files and your favorite text editor) take up no more than one disk of 360K. That leaves enough space for about 275 messages of 20 lines each on your second disk drive. A Fido system with all network material, including a list of over 1500 Fido electronic addresses, leaves space for about 75 messages. (A network of that size takes about 300K, the disk for which needs to be in the second drive because of space considerations.) If you're going to run a full FidoMail system addressing a huge number of nodes, you'll probably want a hard disk. Luckily, hard disks are cheap these days.

## What You Must Already Know

You don't have to be a Grand Hacker to run Fido. It's not all that complicated (although it does take some getting used to), and usually typing a questionmark and pressing Enter gets you help. But you need to know some things about MS-DOS and your computer before you start. Here's a list:

- Operate elementary MS-DOS commands including DIR, COPY, MD, CD, and TYPE
- Specify a pathname
- Get to subdirectories
- Operate a word processor or text editor to make or change ASCII files
- Format disks (if you have no hard disk)

For sophisticated automatic operations, you'll also want to know how to:

- Create batch files
- Use the MS-DOS ERRORLEVEL command (again, for sophisticated operations)

You'll find information on how to create batch files in your MS-DOS or PC-DOS manual.

This manual also assumes that you've seen a Fido in operation — that is, you've already called into a Fido and basically know what it looks like. It doesn't assume that you understand any of the mechanisms underlying what you saw; just that you've called one up and roamed around a bit.



## What's In This Manual

This manual has four chapters, a lot of appendixes, a command summary, a glossary, and an index. The chapters tell you how to set up and use Fido; the appendixes give summaries and technical information.

Chapter 1, "Setting up Fido," tells you how to get Fido up and running quickly and (relatively) painlessly.

Chapter 2, "Running the BBS," gives details on the day-to-day maintenance and operation of the bulletin board message system.

Chapter 3, "FidoNet," shows you how to operate FidoNet, Fido's store-and-forward mail system.

Chapter 4, "Fido V12 Caller's Guide," tells how you'd use Fido if you were a caller.

Appendix A, "Delta Guide to Version 12," lists the changes in this version of Fido so that current Fido sysops can bring their systems up to date quickly.

Appendix B, "Files and Their Functions," lists all the files that come with Fido or that Fido creates, and tells what each file is for. It also says what files are required in all systems, which ones you change or leave alone, and which files are optional and what happens if you leave them out.

Appendix C, "Modem Basics", discusses how Fido and modems interact.

**Traditions abolished and traditions maintained:** The careful reader has already noted that this manual does *not* contain an ASCII chart. Every technical manual in the history of computing has had an ASCII chart in it; Fido Software felt it was time for a bold, perhaps radical, step forward in the art and science of documentation. For our more conservative readers who might be frightened by this apparently reckless move, however, we have included a page that is intentionally left blank. Discovering the location of the blank page is left as an exciting exercise for the reader.

## A Plea For Feedback

We believe that this manual is complete and accurate (although we apologize for the less than exhaustive index). We welcome any comments you have — inaccuracies we missed, typos, things we left out, things that you would have liked to have seen included, and so on. Please send feedback to:

Fido Software  
System Operator's Guide  
164 Shipley Street  
San Francisco CA 94107

Or send us FidoMail:

Scot Kamins  
IFNA Network address 1:125/111

## Chapter 1- Setting Up Fido

Fido comes almost ready to use. ("Little assembly required! Even a child of 5...") If you follow the instructions in this chapter by the numbers, from start to finish, you'll have a functional Fido system up and running very quickly. One of the great things about Fido is that you can change it any time you want without hurting any existing information.

### Hold Your Questions

You might not fully understand some of the things you're going to do in this chapter until you've read the rest of the guide (especially Chapter 2, "Running the BBS"). But that fits in with the goal of this chapter — to get you started as quickly as possible with an up-and-running Fido system. So please hold your questions and curses for a while. All will be answered and made clear in the fullness of time. If it's any comfort, you can at any time change any system parameters that you set up now without harming any information callers might have typed in; so you don't have to worry about messing things up.

If you've still got questions even after you've set up Fido and read the recommended sections, send them to Fido Software via FidoMail (Scot Kamins, IFNA network address 1:125/111). You'll have an answer within a few days.



## First Steps

Here are the steps to take to set up the environment Fido needs to run in. Do the steps in the order listed and you won't get into trouble.

**Current Fido System operators:** Appendix A, "Delta Guide to Version 12", lists the changes in this version of Fido so that you can bring your already on-line systems up to date quickly. Skip the rest of this section and go there now.

**1. Back up the master disk.**

Then put the master away and use your backup for the rest of the steps. That way if something goes wrong you'll still have the master intact.

**2. Decide what your board is for.**

Here are some issues you ought to consider before you go on:

- What's the point of this board?
- Is the board public or private?
- What are the message areas for? How many do you need?
- What about file areas — do you want any? If the files are computer-specific, what computers are they for? Should you have a separate files area for each computer?
- Will your board be for everybody, or just a limited few? Will everybody who uses your board be able to look at all the message areas and all the files? Do you want to restrict who can use certain commands?
- Will you have a FidoNet message and file section? Can anybody use it? Is your mail network public or private?

**3. On a bootable disk (preferably a hard disk), create a subdirectory for Fido.**

Most system operators like to call that subdirectory "Fido". It's not very creative, but it has a certain ring of clarity.

**4. Copy all the files from your Fido backup disk to the Fido directory.**

If space is a problem, just copy the files on the Fido disk's root directory. To see a list of all supplied files, turn to Appendix B, "Files and their functions".

**5. Copy your favorite word processor or text editor to your Fido directory.**

You'll need it to create or change bulletins, help files, and other ASCII text files.

**6. Use your text editor to print the Read Me text file.**

This is a catch-all file that holds last minute news about the system, additions or changes to this documentation, and so on. And even though we don't make mistakes, if we did make them, this is where we'd admit to them.

## Customizing Fido's Files

You need to customize a few files so that Fido can know how to operate your equipment and how to respond to callers, and so that callers can know what to do when they call your system.

This stuff is really simple, and you can use Fido's preset values in nearly every case if you're not sure what to do. (Experience will teach you what you can't figure out now, and you can always change values later without hurting any information already typed into the system.)

### The .INI Files: Fido's Start-up Material

You use the four .INI text files (FIDO.INI, AREAS.INI, COMMAND.INI, and EVENTS.INI) to tell Fido what values to use for all the system parameters — caller download and time limits, DOS ERRORLEVEL values, modem information, FidoNet parameters, and so on. (Technically, a program called SET-FIDO.EXE reads the information you typed into the .INI files, and translates that information for Fido's use. More on SET-FIDO.EXE later.)

**Veteran system operators take note:** The .INI files replace many of the system operator commands and the old command line switches from earlier Fido versions. The product hasn't lost any of its old flexibility or power; the commands have merely been simplified and moved. The benefit is in smaller code size, increased flexibility for future revisions, fewer commands to have to remember, and easier start-up for new people. See Appendix A for details on all the changes to Fido since Version 11.



## FIDO.INI

Open the file Fido.INI with your text editor and change whatever values that you want or need to. You can safely ignore most of the variables for your initial installation—Fido's default values will do nicely. But you have to deal with some of them.

Assuming you're running a full Fido system (BBS and FidoNet), you should at least consider the values for variables marked with √ in the chart below. Fido doesn't care whether you use capitals or small letters, and the order that the variables appear in doesn't matter. But be sure to spell a variable name exactly as it appears, or your system will throw up. A vertical bar (|) means that any value on the left or the value on the right is valid for the variable in question.

The column marked **Variable** in the following chart is the variable name that appears in Fido.INI — don't change this name or the program won't run properly. **Default** is the value that Fido uses for the variable if you don't change it. **Range** shows the alternatives and limits you can use for a given variable. **Old Command** is for veteran Fido system operators updating their systems — if you're new to Fido, you can ignore this column. **Meaning** is an abbreviated description of the variable. (Following the chart you'll find a detailed explanation for each variable.)

<p><b>If you don't get it, don't mess with it:</b> Don't change the value of a variable that you don't understand. Fido assigns the safest, most common value to each variable; don't experiment until you've read and understood the longer explanations following the chart. This is especially important for the variables Node-Number, Net-Number, Zone-Number, Alt-Node, Alt-Net, Alt-Zone, and all ErrorLevel variables (which are, by definition, for experts only).</p>
---

# Table of Preset FIDO.INI variables

Variable	Default	Range	Old Command	Meaning
<b>Time and download limits</b>				
Signon-Limit	10	1-∞ minutes	/S	Time to complete sign-on
First-Limit	30	1-∞ minutes	/F	First time caller limit/call
Normal-Limit	60	1-∞ minutes	/L	Normal caller time limit/call
Daily-Limit	180	1-∞ minutes	/D	Total limit/48 hours
K-Limit	200	1-∞ Kbytes	/K	Download limit/48 hours
<b>DOS ERRORLEVEL values</b>				
E-ErrorLevel	0 [off]	3-255	/W	E command ERRORLEVEL
O-ErrorLevel	0 [off]	3-255	/A	Caller O)utside ERRORLEVEL
G-ErrorLevel	0 [off]	3-255	/E	What happens at G)oodbye
Mail-ErrorLevel	0 [off]	3-255		Handling unscheduled FidoMail
<b>Modem Information</b>				
√Modem-Type	1	1-14	/J	Type of modem
√CD-Bit	128	8 16 128	/V	Carrier Detect (CD) bit value
Modem-String	[nil]	1-40 characters		Special modem instructions
IO-Port	1	1 2	/1,/2	Select Async port COM1 or COM2
<b>FidoNet Parameters</b>				
Connect-Tries	1	1-255	/G	Times/schedule Fido tries to call a node, connect but no mail sent
Dial-Tries	20	1-255		Times/schedule Fido tries to call a node
Node	0		4	This node's number
Net	0		4	This net's number
Zone	0			This zone's number
Alt-Node	0	0 Node	4	Alternate for Host node; else same as Node number
Alt-Net	0	0 Net	4	Alternate for Help net or private net; else same as Net number
Alt-Zone	0	0 Zone		Alternate for Host Zone number; else same as Zone number
√Money	\$	<character>		Money character
Thousands-Separator,		<character>		As in 19,200 baud
Decimal	.	<character>		As in 3.14159
<b>Miscellaneous Parameters</b>				
√Private	No	Yes No	/M	Private system?
Msg-Length	100	1-175		Lines/message
Default-Priv	2	0-4 7	/P	New caller gets privilege level 2
New-Caller-Key	[none]	1-32		Optional key for new caller
Path-Display	No	Yes No	/Y	Show path info in A)rea command?
Xfer-Display	No	Yes No	/U	Display file transfer status on system operator's console?



## Time and Download Limits

**Signon-Limit:** The amount of time in minutes a caller can take to complete the sign-on. Timing starts as soon as Fido detects a carrier, and continues until the caller enters a valid password. Please be sensitive with this one; some physically disabled callers might be using special equipment to call your system. It might take them longer than you might expect to read your new-caller messages (explained later in this chapter) and to respond appropriately.

**First-Limit:** The amount of time in minutes first-time callers can stay on the system. This only applies to the first time they call.

**Normal-Limit:** The amount of time, in minutes per call, callers with passwords can stay on the system. Enter a 0 for no time limit.

**Daily-Limit:** The amount of time in minutes per 24-hour period callers with passwords can stay on the system. Enter a 0 for no time limit.

**K-Limit:** The number of bytes in kilobytes a caller can download from the Files area in a 24-hour period. The kilobytes a caller uploads impacts this number, increasing it by twice the number of bytes s/he uploads for any given 24-hour period. (For example, if the number here is 200 and a caller uploads a 300K file, s/he gets  $200 + (300 * 2)$  Kbytes, or 800 Kbytes, to download. When s/he uses up this bonus, s/he goes back to the K-Limit. Enter a 0 for no limit.

## DOS ERRORLEVEL Values

Fido uses DOS ERRORLEVEL protocols for several purposes, most of which have to do with scheduled external events (such as running housekeeping programs or FidoMail). Four of them are listed here because they potentially impact every call.

**E-ErrorLevel:** The ERRORLEVEL that Fido sends to DOS when a caller uses the E)xit DOS command to terminate Fido and exit to DOS. Setting this variable to 0 (zero) disables the command. More often than not, it exits to a batch file that lets the system operator run the system remotely.

**O-ErrorLevel:** The ERRORLEVEL that Fido sends to DOS when a caller uses the O)utside command to terminate Fido and exit to DOS. You should set the privilege level for using this command very high. It's a good idea to let somebody use this command only if Fido exits to a batch file you've already created; otherwise you're giving the caller access to your DOS as if s/he were sitting at your console! Setting this variable to 0 (zero) disables the command. See "Setting Access Levels" later in this chapter for more details.

**G-Errorlevel:** The ERRORLEVEL that Fido sends to DOS when a caller exits normally (the G)oodbye command) or when s/he just hangs up. Take care with this one; if you use any number but 0 for this variable, Fido will terminate after every call and won't restart automatically.

**Mail-ErrorLevel:** The ERRORLEVEL that Fido sends to DOS when unscheduled FidoMail arrives. When this value is set to 0, Fido accepts the incoming mail and handles it the way it normally would. When this value is set to nonzero, Fido exits to DOS, presumably to a batch file running a special program to handle incoming mail in some nonordinary way.

## Modem Information

These variables give Fido the information it needs to determine what codes to send to your modem, when to send them, and what to expect back.

**Modem-Type:** A number identifying the brand of your modem, telling Fido how to dial and disconnect, the modem's maximum baud rate, and other important information. Find your modem in the following list and use the listed value. If your modem isn't here, try using number 1; if that doesn't work, replace the modem with one on this list:

DC Hayes Smartmodem 1200 and clones	1
DC Hayes 2400	1 1
Multi Tech 2400	1 3
Prentice POPCOM	7
US Robotics 212A	5
US Robotics Courier 2400	1 2
US Robotics Courier HST 9600	1 4

**CD-Bit:** Value of the mask for the serial interface status port when the carrier detect line (CD) goes high, indicating that it hears a modem carrier on the line. If you don't know what this means, don't worry about it; just find your computer on the following list and use the listed value:

IBM (all models and clones)	1 2 8
DEC Rainbow 100	1 6
Victor 9000	8

**Modem-string:** Special instructions your modem might need to see if it's not one of the recommended modems, or if your installation is unusual. Fido sends this string to your modem after Fido has completed its normal initialization.

**IO-Port:** Normally Fido assumes you want to use COM1 as the serial port. Change this to 2 to make Fido use COM2.

## FidoNet Parameters

These variables deal with sending and receiving electronic mail. If you're running your Fido just as a bulletin board system, you can skip these commands entirely. For complete details on the mail system, see Chapter 3 and the section in Chapter 4 called "Sending and Receiving FidoMail".

**Dial-Tries:** Number of times your system will try calling a particular node in a given schedule (assuming there's enough time). If for some reason Fido can't get through to a particular node the first time it tries (usually because of busy signals, less frequently because a node is down), that node's number moves to the bottom of Fido's calling list for that schedule. After Fido makes the rest of its calls, it again tries calling the node or nodes it missed. This process continues until (1) all nodes are called successfully; (2) the number of tries for a particular node meets the value in Dial-Tries; or (3) time allotted for the scheduled event runs out.



**Connect-Tries:** Number of times in a given schedule your system will recall a particular node to re-establish an initially faulty connection. When a faulty connection happens with a node, that node's number moves to the bottom of Fido's calling list for that schedule. After Fido makes the rest of its calls, it again tries calling the node or nodes it missed. This process continues until (1) all nodes are called successfully; (2) the number of tries for a particular node meets the value in Connect-Tries; or (3) calling time runs out.

**Node:** The number your system has been assigned by the administrator of your network. Fido uses this to make sure incoming mail really belongs to your system, and to put a return address on outgoing mail. *Don't change this number on your own, even temporarily.* Contact the network administrator to get a number.

**Net:** The number of the net to which your node belongs. *Don't change this number on your own, even temporarily.* Contact the network administrator to get a number.

**Zone:** The number of the zone to which your net belongs in the world-wide FidoNet system. *Don't change this number on your own, even temporarily.* Contact the network administrator to get a number.

**Alt-Node:** Your other node number, if it exists. Sometimes a node can have two numbers. For example, if your system is the host node of a net, it will have its own number (the Node-Number), plus its host number (always 0, its Alt-Node). *Don't change this number on your own, even temporarily.* Contact the network administrator to get a number.

**Alt-Net:** Your other net number, if it exists. Sometimes a node can be in two nets at once. *Don't change this number on your own, even temporarily.* Contact the network administrator to get a number.

**Alt-Zone:** Your other zone number, if it exists. Sometimes a node can be in two zones at once. *Don't change this number on your own, even temporarily.* Contact the network administrator to get a number.

**Fido Software doesn't assign Nodes:** Fido Software doesn't assign node, net, or zone numbers. If you're part of a private network, contact your network administrator for number assignments. If you want to be part of the public International FidoNet Association network (IFNA), contact that organization for numbers. You can write to them at P.O. Box 41143, St. Louis, Mo, 63141 or send them a piece of FidoMail (using a FidoMail system that's already registered, of course) at IFNA network address 1:1/0.

**Money:** What character to use for currency displays. Fido uses currency displays when it shows how much credit a caller has for the cost of sending mail. (You determine how much it costs to send a piece of mail, if anything; see Chapter 3 for details.)

**Thousands-Separator:** The character that Fido uses to separate numbers into groups of three (as in \$12,500). For example, in the United States you'd use the comma (,); in many European countries you'd use the period (.). You can't use the space character.

**Decimal:** The character that Fido uses to separate whole numbers from fractional parts. For example, in the United States you'd use the period(.); in many European countries you'd use the comma (,). You can't use the space character.

## Miscellaneous Parameters

These parameters affect various parts of the system.

**Private:** Determines whether your system will be public (anybody can call and apply for a password) or private (only people who already have passwords can call). You can make your system semi-private if you want; assign Private the value Yes, and see "Public vs Private Systems" in Chapter 2.

**Msg-Length:** Number of lines a message can be. By keeping this number fairly large, callers can enter essays and long opinions; this encourages in-depth discussions of issues. The size of the message you allow should be determined by the nature of your board.

**Default-Priv:** Sets a privilege level in the range 0 through 4. You can use this, for example, to provide system security while you evaluate someones application for membership to your board. See "Setting Access Levels" later in this chapter and "Privilege levels" in Chapter 2 for more details.

**Keep this level low!** It's important to keep the default privilege level below 4. Level 4 gives callers access to FILES.BBS (see Chapter 2) and the ability to do File Attach while sending a FidoNet FidoMail message. If a caller knows pathnames in your system, s/he can attach any file or files to a message. The bottom line here is that a caller with Privilege Level 4 who also knows about your pathnames can steal the entire contents of your disk and send it to the other side of the world at your expense.

**New-Caller-Key:** Optional variable to give a key (numbered between 1 and 32) to new callers. You use keys as a way of further limiting access to commands. You can assign more keys to new callers by adding an additional **New-Caller-Key** line for each key. See "Setting Access Levels" later in this chapter for more information about keys.

**Path-Display:** Determines whether Fido shows path information in the A)rea commands, in addition to what you've typed into each area's description in the AREAS.INI file (described elsewhere in this chapter). This information can be sensitive, especially if you're letting people use the O)utside command to get access to your system, in that it tells callers about your file system's structure.

**Xfer-Display:** Determines whether Fido shows file transfer status on the system operator's console. This is of interest only to people with privilege level 7, the highest level. Set to Yes, this lets Fido display status information about each block being transferred, including any errors that occur. Because it takes time to display the information, data transfer is slowed slightly at 9600 baud and above.



## AREAS.INI

The AREAS.INI text file defines Fido's message and file areas. (The areas are actually disk subdirectories). Use it to set up all file and message areas, including

- the message and/or file area that new callers use
- the area where messages left for the system operator at the G)oodbye prompt go
- the areas for FidoMail files and messages.

You also use AREAS.INI to assign privilege levels and locks to file and message areas.

### Defining an Area

You define an area by entering a command line. Here are some typical ones:

MsgArea=C:\GenMsg	D="General Messages"	O=NewCaller
MsgArea=C:\Fido\FidoMail	D="Mail Call"	O=FidoNet
MsgArea=OffNotes	D="Caller Questions"	O=Goodbye
FileArea=MemFiles	D="Hot Software" P=4	L=1,3,5
FileArea=NewRevs	D="Caller-Sent Software"	

Each command line defines a single area and is made up of a series of expressions. The first expression must be either `MsgArea=pathname` or `FileArea=pathname`, defining this as a message area or a file area. Next on the line is an expression describing the area, and one or more optional expressions. Here are the rules:

- Fido doesn't care if you use capitals or small letters.
- Each expression must be separated from other expressions by either a space or a tab character.
- No space or other character can come between the equal sign and either side of the expression.
- The entire command line must fit on one screen line (no line feeds) with no imbedded carriage returns.

**Pathname is subdirectory:** The pathname is in reality an MS-DOS subdirectory name. Fido will create this directory for you if it doesn't already exist.

The following list shows what expressions you can use, and what each expression means.

**MsgArea=pathname:** This expression starts the definition of a message area. You can have up to 200 message areas; most folks find more than 10 to be unwieldy.

**FileArea=download pathname:** This phrase starts the definition of an area holding files that a caller can download. (See the note under **U=upload subdirectory**, below.) As with message areas, you can have up to 200 separate file areas.

**D="description":** This required expression briefly describes the area. The description appears when a caller chooses the A)rea command and becomes part of the location prompt when s/he enters an area. You must enclose the description in quotation marks if you want to imbed spaces (as in "General comments"). The description can be up to 63 characters long.

**U=upload subdirectory:** This optional files subdirectory receives any files that a caller uploads. It's usually best to create a separate subdirectory for uploaded files so that you can

examine them and confirm their appropriateness for your system before you make them available to your callers. If you don't specify a U directory, nobody can send any files to this area. This area can be the same as the **FileArea** subdirectory: the advantage of having both the **U** and **FileArea** subdirectories the same is that maintenance is easier — if the **U** directory is different, you have to move files from **U** to the **FileArea** subdirectory to make them downloadable; the disadvantage is that illegal or immoral files can be dispersed to the unsuspecting public before you have a chance to call your lawyer. (Also see "Using @ to Limit New File Display" in Chapter 2.)

**Separated upload area important:** If you're running a public bulletin board, give strong consideration toward creating a separate area to hold files uploaded by callers. Callers sometimes think a program is in the public domain when it actually isn't, and they upload the program with the best of intentions. As the operator of the board, you are legally responsible (as in *law suit, fine, jail*, etc.) for files that people can download. So it's in your best interest to check the copyright and ownership of any file uploaded to your board before you make it available for downloading.

**P=*privilege level*:** Use this expression to limit who can use the area. *Privilege level* is a digit in the range 0 - 4 (or 7 for system operators). Only someone with a privilege level at least as high as the one set here can use this area, assuming s/he has the necessary key(s). (See **L=*lock***, immediately following this paragraph.) You set privilege levels for first-time callers through the Fido.INI file (described earlier in this chapter), and for more experienced callers through the SYSOP.EXE utility program (described in Chapter 2). If you don't include a privilege expression on the command line, Fido sets a default privilege level of 1.

**L=*lock*:** Use this expression to further limit who can use the area. *Lock* is one or more numbers in the range 1 - 32, separated by commas with no imbedded spaces. Only someone with key number(s) matching the one(s) set here can use this area, assuming s/he has the necessary privilege level. (See **P=*privilege level***, immediately preceding this paragraph.) You issue keys to callers through the SYSOP.EXE utility program (described in Chapter 2). If you don't include a lock list on the command line, Fido assumes that any caller with the appropriate privilege level can use the area.



**O=option:** Use this expression to set one of a series of options that further define an area's use. The chart below shows each option and its uses. You can type all the characters in the entire option, or only as many characters as is necessary to make the option unambiguous (for example, N for the NewCaller option).

Option	Shortcut		Area	Type	Default	Description
Goodbye	G	MsgArea	No prompt			Enables Goodbye prompt; tells where to put message when caller leaves one for system operator at Goodbye prompt
FidoNet	F	both	First defined			Where FidoNet messages and files go
NewCaller	N	both	First defined			Where first time caller goes when s/he chooses F)ile-Section or M)sg Section on the MAIN menu
Overwrite	O	FileArea	Can't overwrite			Allow newly uploaded file to replace file in Upload area with same name

You can have more than one option on a command line, so you can do things like have the same area be the default area for first-time callers, as well as the area that holds G)oodbye messages:

MsgArea=General      D="General Messages"    O=NewCaller      O=Goodbye

**Remember the Mail:** FidoMail can't work if you haven't defined areas for mail messages and for mail files using the FidoNet option. If you plan to send or receive mail, be sure to set up these areas.

# COMMANDS.INI

The COMMANDS.INI text file defines the text of all Fido commands as they appear to callers. It also defines the privilege levels and optional lock-and-key combinations necessary to see the commands. (A command you can see is a command you can use.)

The command name, privilege level, and lock list (if you use one) must all fit on one screen line (no line feeds) with no imbedded carriage returns. Use spaces to separate elements on a line:

F)file-Section

P=2

L=3,5,7,8

**Don't change what you don't have to:** Thousands of Fido systems exist with the same command names. This makes it easy for people to use them — they know what to expect. If you don't have a really good reason for changing a command's name, please don't do it.

## Changing Command Names

You might find it necessary to change the text of Fido command names. For example, you might want to localize the system for non-English speakers. When you change the text, follow the rules and suggestions listed here. The first seven items describe how the system works, and you must follow them:

- **Order important:** Command names must be given in the order listed for each section (Main, Files, Messages, etc.). Fido reads the order of commands — the first command enters the message section, the second enters the file section, and so on, *no matter what name you give the command*. There is no way to change the order of commands. You can, however, hide a command by setting a high privilege level, or by giving it one or more locks.
- **First letter is command:** The first letter of the command name is the actual command that the caller types. The entire command name appears when the caller chooses the novice help level, All (the preset level).
- **Commands start with unique alphabetic characters:** Each command name in each section must start with a unique visible alphabetic character (capital or small letter). If more than one command name in a section starts with the same character, when somebody logs in Fido will use the earliest one in the section and ignores the duplicates. Technically you might get away with using control characters, *but future versions of Fido might use them for other purposes and your system might not function properly*. The system won't stop you from using digits and math symbols, but Fido uses these in other places; the resulting competition for meaning will cause confusion (at best), and the command might not work. The space character, tab, comma, and semicolon are used to separate command elements (and for other arcane uses) and are therefore not allowed.
- **Command is one word long:** A command name is always a single word (no imbedded spaces or tabs). If you want a space to appear in a command name, use the underscore character (`_`); Fido will convert it to a space for you at display time. You can also use hyphens and capitalization to fake multi-word commands. For example, you can use Area-Change and ExitToDos.



- **20-character limit:** Command names can be up to 20 characters long. All characters after the 20th are dropped.
- **? gets help:** The questionmark/slash key always gets help. You can't change this.
- **Semicolons start comments:** Fido converts the .INI files to machine-readable code when you run the program called SET-FIDO. (In technical terms, it *compiles* the code.) To the computer, a semicolon is a special symbol that means "Everything from here to the end of the line is for humans, not for me". If you want to add notes to yourself, start the note with a semicolon. If the note wraps to a new line, start the wrapped line with a semicolon. By the same token, beware of spurious semicolons; the computer ignores everything after it. Conversely, if you take out a semicolon, be sure to remove the note that follows it; Fido will try to compile everything that *doesn't* start with a semicolon, and may get sick if it digests random characters.

Here are a few suggestions you really ought to follow. The system will probably work if you ignore them, but callers might send you letter bombs:

- **Check all Help files:** If you change the command G)oodbye to A)u\_Revoir, be sure to also change the name in the descriptions you include in Help files. Any slipups are bound to confuse callers.
- **Be consistent:** Nothing will drive a caller to distraction faster than inconsistencies in command names. For example, if Q means Q)uit in one place, don't use L for L)eave in another. Further, don't have G mean G)oodbye (as in Log Off) in one menu, and have it mean G)o\_To\_Main\_Menu someplace else.
- **Check all commands and prompts:** Because you're making changes, it's your responsibility to check every command, prompt, and help level throughout the system to make sure that everything works and that everything matches.

## Setting Access Levels

Access levels determine who can use which commands (and, as you've already seen, which message and file areas). A caller's access level is determined by a combination of the caller's privilege level and what keys the caller has, and how that privilege level and key combination matches the demands of the system function s/he wants to use.

You assign the privilege level and keys to a specific caller by using the Sysop.EXE program, described in Chapter 2. You assign a default privilege level and default keys for new callers through the FIDO.INI file, described earlier in this chapter.

Each command, message area, and file area can have limited access. If a command has a privilege level of 3 attached to it, a caller must have a privilege level of 3 or higher to use that command. If the command also has locks attached to it, a caller must have matching keys (in addition to meeting the privilege level restriction).

### About Privileges

You set privilege levels to restrict the use of more sophisticated commands (ExitDos or Override-Path), to reserve use of potentially sensitive areas (List-Callers), or perhaps to give privileges to subscribing members (File-Section).

Each command has a preset privilege level from 0 through 4 (or 7 for "system operator only" commands). If a command's level is the same as or lower than the privilege level of a caller, the command is visible to, and therefore usable by, the caller. (If you can see a command in Fido, you can use it.)

To change a command's privilege level, change *n* in each command line's *P=n* expression:

C)hange      P=1

### About Locks

You can further restrict access to a particular command by adding locks to it. You might want to do this to differentiate among groups of callers, all of whom have the same privilege level. For example, 200 callers might all have privilege 3, but you want to let only 50 of those callers use the Files section.

You assign a lock or a list of locks by adding to a command line the expression *L=n1[,n2,...n32]* where *n* is a number between 1 and 32. For example:

F)ile-Section      L=28

A caller needs key number 28 to get to the Files section.

Lock numbers are not hierarchical, and they are separate from privileges. No lock-and-key combination has a higher value than any other combination; the important thing is that lock and key numbers match. If somebody has a privilege level of 4 but holds no keys, s/he won't be able to use a command that has a lock on it, no matter how low its privilege level is.



## Table of Preset Command Names and Privileges

Here's a list of the commands in the system showing their preset privilege levels. (There are no default locks.)

Command	Default privilege	What command does
<b>Main Section commands</b>		
M)sg-Section	0	Enter message section
F)ile-Section	1	Enter file section
B)ulletins	0	Display bulletins
S)tatistics	0	Display statistics
C)hange	1	Change caller settings
N)ame	2	Change caller name
A)ddress	2	Change caller city
P)assword	0	Change caller password
H)elp-Level	0	Change caller help level
W)idth	0	Set screen width (columns)
L)ength	0	Set scren length (lines)
M)ore	0	Toggle More prompt (on off)
T)abs	0	Toggle tabs expan. (on off)
F)iller-Nulls	0	Set nulls
P)age-Operator	1	Page the sysop
L)ist-Callers	2	Display the caller list
A)ns-Questionnaire	0	Fill out a questionnaire
V)ersion	0	Display Fido version
O)utside	7	Drop to MSDOS (callers)
E)xitDOS	7	Exit to MSDOS (sysop)
G)oodbye	0	Log off the system

### Message Section commands

N)ext	0	Read higher message
P)revious	0	Read previous message
E)nter	2	Enter a new message
K)ill	2	Delete current message
T)o-You	0	Check for mail
A)rea-Change	2	Select a message area
R)eply	2	Reply to current message
C)ontinuous	0	Read continuous until abort
L)ist	0	List message headers
S)earch	0	Search header contents
O)verride-Path	7	Set a specific pathname
G)oodbye	0	Logoff
M)ain-Menu	0	Quit to Message Section

Command	Default privilege	What command does
<b>File Section commands</b>		
F)iles	2	List files in this area
D)ownload	2	Download a file
U)pload	2	Upload a file
A)rea-Change	2	Change file area
T)ype	3	Type text file
S)tatistics	0	Display statistics
L)ocate	2	Search all file areas
R)aw-Display	3	Display complete file info
K)ill-File	7	Delete a file
O)verride-Path	7	Pathname override
G)oodbye	0	Log off the system
M)ain-Menu	0	Return to main section



## EVENTS.INI

Unless you tell it differently, Fido is always ready to carry out its common functions: let human callers read and write messages or download and upload files; and let computer callers pick up or leave electronic mail. An **event** is something special that Fido does starting on a particular day at a particular time (or within a particular period of time). EVENTS.INI is the place where you plan such events.

**Attention scared system operators:** You don't need to have any events defined at all in order to use Fido. If you don't want to run any events, ignore this section of the manual. But you must set an event if you want to let callers use the P)age-Operator command, or if you want to use FidoNet, the electronic mail program.

### Event Line Components

Each event definition, which is always made up of the four components shown below, must fit on one line. The components of the event line must be in this order:

*DAY*                      *TIME*                      *WINDOW*                      *EVENT*

(You can use tabs or spaces to make the file more readable.) You can schedule up to 100 separate events, as long as each has its own event line.

*DAY* is the day of the week the event should run. Days are named MON, TUE, WED, THU, FRI, SAT, SUN and ALL. ALL means every day (which saves you the trouble of entering seven nearly identical events when you want the same thing to happen every day).

*TIME* is the time of day the event should start in hours and minutes, using a 24 hour format. The form is MM:HH, where 00:00 is midnight, 12:00 is noon, and 23:59 is one minute before midnight (11:59 PM). *TIME* isn't necessarily the time an event *must* start, but rather the time at which the event *may* start. The event may start anywhere within the event's *WINDOW*, described next.

*WINDOW* is the number of minutes from the start time (*TIME*) within which the event can start. (Sometimes a previously started event is still running when the start *TIME* for the new event has come.) For example, if you want an event to start somewhere between 9 AM and 11:30 AM, you'd use a *TIME* of 9:00 and a *WINDOW* of 150 (2 hrs 30 minutes is 2 \* 60 plus 30). If the event is anything except an ERRORLEVEL, then *WINDOW* also defines the length of time the event can run — when the number of minutes in *WINDOW* has expired, the event ends.

*EVENT* is the event itself. Fido recognizes four types of events: **Page**, defining when a caller can use the P)age-Operator command to summon the system operator; **FidoNet**, defining when Fido sends electronic mail to other Fido nodes; **Idle**, defining when Fido does nothing (which includes not accepting calls); and **ErrorLevel**, defining when Fido turns over control to MS-DOS (usually back to the batch file that originally executed Fido).

And now the details:

**PAGE:** During the time PAGE is active, the P)age-Operator command will try to get the attention of the system operator, who can talk to the caller via the local keyboard. The example enables the P)age-Operator command from 9:00 AM until 5:00 PM:

**FIDONET *schedule tag*:** This event runs FidoNet, the electronic mail system, during the specified time. *schedule tag* is a letter from A through W. For details, see Chapter 3. The example has Fido sending mail at 2 AM for one hour:

ALL 2:00 60 FidoNet A ;Typical IFNA National Mail Hour

**IDLE:** Use IDLE when you want Fido to do nothing but spin its wheels. IDLE is especially useful when you're sharing a single phone line between a modem and a voice telephone, and you want to stop Fido from answering the telephone during business hours:

WED 8:00 600 IDLE ;Fido won't answer the phone 8 am - 6 pm

**ERRORLEVEL *number*:** ERRORLEVEL makes Fido terminate to DOS with the specified ERRORLEVEL (a number from 3 through 255 inclusive). Usually you'll use ERRORLEVEL only when you've executed Fido from a batch file. (You can test for an ERRORLEVEL number from within that batch file). For details on both ERRORLEVEL and batch files, see your DOS manuals.

MON 4:00 5 ERRORLEVEL 17 ;Exit to DOS at 4:00 AM on Monday

**Watch your windows:** ERRORLEVEL leaves Fido and exits to DOS, most likely to a batch file. In ERRORLEVEL event lines, *WINDOW* only means the number of minutes from *TIME* that the event can start; it does not control how long the event can run because Fido is no longer in control of what happens. This brings up another issue: you need to be sure that your next event has enough time to run in case Fido starts later than you think it will. In practice, do several practice runs with all ERRORLEVEL events to see how long they take, and give a wide window to an event scheduled to start following an ERRORLEVEL event, in case the ERRORLEVEL event takes longer to complete than you anticipate.

## The Special OPT Keyword

Normally, Fido will limit how long a caller can be on line so that s/he doesn't interfere with an upcoming event. For example, if an event is scheduled for 9:00 AM and a caller logs on at 8:45, Fido allows the caller only 15 minutes, no matter what the normal time limit is, to ensure the system will be free at event time. (Fido being the warm and fuzzy system that it is, the caller gets appropriate warnings about the temporary time limit.)



Sometimes you'll decide that it's OK for a caller to preempt a scheduled event. For example, you need to run a program via an ERRORLEVEL event sometime during the day, but you don't care what time it starts. In these situations, start the command line with the special keyword OPT (for OPTIONAL). OPT tells Fido not to cut a caller's time limit short. When the caller hangs up, Fido runs the event — assuming, of course, you've set the event's window wide enough to account for the longest possible time limit. Here are two examples:

```
OPT ALL 8:00 240 ErrorLevel 8 ;execute when no user is on
OPT ALL 8:00 10 ErrorLevel 9 ;could be missed entirely
```

As an exercise, assume that no caller has a maximum time limit longer than 4 hours: In the first example, if the caller logs in at 7:59 and stays on for 3 hours 55 minutes, the event will start at 11:54. No matter what time a caller logs in, the event will run.

In the second example, if a caller logs in before the event and stays connected between 8:00 and 8:20, the event won't happen at all — a true case of an OPTional event.

## Event Execution

Fido searches the event list from top to bottom, and executes the first event line that's executable — that is, the current time is within that event line's window — and that hasn't been run yet. (Fido keeps track of event lines that have already run on a given day and won't run them again.)

You can schedule an event to run at different times on the same day, as long as you use separate event lines to do it. The following example has the P)age-Operator command in effect daily from 9 AM to Noon, and from 1 PM to 5 PM (under the theory that even Fido operators eat lunch):

```
ALL 9:00 180 PAGE ;enable PAGE (9 AM - Noon)
ALL 13:00 240 PAGE ;enable PAGE (1 PM - 5 PM)
```

You can also specify single-day exceptions to a schedule, as long as the exception runs longer than does the regularly scheduled event:

```
ALL 9:00 480 PAGE ;enable PAGE (9 AM - 5 PM)
SAT 8:00 600 PAGE ;enable PAGE (on Sat 8 AM - 6 PM)
```

**No bad puns:** A certain Fido programmer and copyright holder has suggested that the author of this manual has the opportunity to give the example of a PAGE event which is enabled at 6:00 AM; thus the author could abuse his readers by talking about an event you "execute at sunrise." The author, however, is above such shabby punning. The author also figures that this means you, the reader, owe him one. Watch for it.

## Customizing System Notices

Fido uses text files to hold information a caller sees from time to time as the program runs. Most of these system notices are in help files that only appear when a caller asks for them. But others are system notices that appear automatically in different parts of the program. You might want to modify these text files to go with the theme of your board.

**Killing CONTROL-C:** Ordinarily a caller can abort a system notice by pressing CONTROL-C or CONTROL-K. You can stop CONTROL-C or CONTROL-K's effectiveness by embedding a CONTROL-B in the notice. A caller won't be able to cut short the notice until the end of the notice, or until Fido comes across an embedded CONTROL-C.

Here's a list of the files whose contents you should be sure are appropriate for your board before you go on-line for the first time. You can modify them in your text editor.

**Follow along in your computer:** The rest of this section will make more sense to you if you follow along in your computer. Use your text editor to look at each file so you can see what this section describes.

**WELCOME1.BBS:** This is the first text a caller sees. Use it to give the name of the board, the board's purpose, and the name or handle of the system operator (so people can leave you messages).

**WELCOME2.BBS:** Text from this file appears after a caller with a password logs in. Use it to give callers news or other information. Many systems run without this file. If you're not going to change it, delete it; the information it contains will be meaningless to callers, and will be a constant source of embarrassment to you and your family for generations to come.

**NEWCLR1.BBS:** On a public Fido, this file's text comes up when somebody whose name isn't known to the system logs in. It appears just after callers type their first and last names. Use it to tell new callers what the board is about, that they'll need to enter the city and state they're calling from, that they'll need to pick a password of up to eight characters, what a password is for, and so on. You really *should* include this file, so that new callers won't be left without a frame of reference. This file's contents only appears the first time a new caller logs on.

**NEWCLR2.BBS:** Fido displays this file's contents right after new callers enter their city and state, and pick a password. Use it to give a simple tutorial of your system, where to find things, what the rules are, and so on. Like NEWCLR1.BBS, this file's contents only appears the first time a new caller logs on.

**BULLETIN.n:** Most system operators use the bulletins (up to 99 of them) to hold frequently-changing information. *n* is a number from 1 through 99. If these files exist, they're listed when a caller enters the B)ulletins command from the MAIN command prompt. Fido lists the first nonblank line in a bulletin's file as the bulletin's descriptor.

**Note to veteran Fido system operators:** The Bulletin.*n* files replace the old Bulletin.BBS file. The old file called Editorial.BBS has been trashed.

**QUOTES.BBS:** Quotes is a file filled with hundreds of quotations from the famous and not-so-famous. Each time a caller logs on successfully, a quotation appears just before the Main



Menu scrolls onto the screen. You can edit this file as you see fit, deleting quotations you dislike or adding your favorites. If Fido can't find this file (when, for example, you don't copy it to your working disk), it pretends the Quotes feature doesn't exist.

**PAGE.BBS:** Callers see this file's contents if they enter the P)age operator command when paging isn't scheduled. Use it to tell callers when you're around to answer pages. If you delete this file, Fido gives a message saying that the system operator isn't available.

**name.HLP:** Fido comes with a complete series of contextual help files: The caller types the first letter of the command name immediately followed by a questionmark and a carriage return to get help on that command. (S/he can also just enter a questionmark and carriage return to get general help). You shouldn't have to change them unless you change the command names in COMMANDS.INI. But just in case, here are the names of each of the help files and what they give help about:

<b>MAIN.HLP</b>	Commands in the Main section
<b>CHANGE.HLP</b>	Items that appear when the Change command is invoked
<b>MESSAGE.HLP</b>	Commands in the Message section
<b>EDIT.HLP</b>	Items that appear in the message editor when a caller finishes typing in a message (Save, Continue, Abort, etc.)
<b>NODES.HLP</b>	Items that appear when a caller starts to enter a message in the FidoNet section (Net, Node, etc.)
<b>FILES.HLP</b>	Commands in the Files section
<b>XFERTYPE.HLP</b>	Items that appear when it's time for a caller to choose a protocol for file upload or download (Ascii, Kermit, Xmodem, etc.)

## How Help Works

When you look at any help file in your text editor, you'll see that it's a collection of separate items. Each item begins with a backslash (\) at the left edge of line, and is immediately followed either by a questionmark or by the first letter of a command (as in \F):

\F

"F)iles" lists all of the files available for download in this File Area. You can list files selectively using MSDOS "wildcards"; "F\*.EXE" lists all files with file type .EXE, etc.

Control-C aborts commands/text display

Control-K aborts commands/text display

Control-S pauses commands/text display

\D

"D)ownload" sends files from Fido TO YOU. You are asked for the file transfer protocol to use first (you can get specific help at that point also) then the file(s) you wish to download. Please refer to the documentation for your terminal program on how to download. You must also complete your download within your time limit left. (Fido will warn you at the right time.)

When a caller types in a letter and a questionmark (as in F?), Fido searches the appropriate help file searching for a backslash followed by the character that the caller has typed, and displays everything between that character and the next line-starting backslash (or the end of the file). If the caller enters just a questionmark, Fido looks for \? in the file, and does the same thing.



## The Questionnaire

Fido lets you create questionnaires for callers. A caller can fill out a questionnaire by using the A)ns-Questionnaire command in the Main menu; or you can make the caller fill one out at sign-on time as a requirement for getting into a private system. Whether you create a questionnaire or not is up to you. (If a caller tries the A)ns-Questionnaire command and no questionnaire exists, Fido says so.)

The questionnaire itself consists of a series of questions in a text file. The questions can be open-ended ("How long have you felt the need to communicate through a computer?") or multiple choice ("I've been avoiding talking to people face-to-face for: (1) less than 5 years (2) 5-10 years (3)..."). Responses to open-end questions can be up to 78 characters long. Multiple-choice questions can have up to nine alternate answers, with an option for an additional answer if the caller chooses the highest-numbered answer. (For example, you might have five possible answers, with the fifth choice being "(5) Other".)

Answers to the questionnaire are appended to the end of a second text file in a fixed-field format. Answers are numbered to match the corresponding question.

**Nonstandard Disclaimer:** The questionnaire is meant to be extremely simple. It was included in the original Fido (Version 1) so that system operators could ask a few simple questions to screen potential applicants to private systems, and it hasn't been improved since then. For some reason, some system operators insist on using the questionnaire for far more complex purposes than it was originally designed. Beyond the format of the answer file itself (explained below), the Fido package contains no method for comparing or interpreting caller's answers. Post-processors are available in the public domain for doing such interpretations, but Fido Software has no information about how worthwhile or reliable they are.

## Constructing the Questionnaire

You construct a questionnaire using your text editor. Fido interprets the questionnaire by looking at the first character of each line. If it finds no special character (as outlined below), Fido displays the contents of the line literally; use this feature for displayable notes, special instructions, or for longish questions. Here are the special characters and what they mean:

- \* Add the caller's name plus the current time and date to the answer file. Use this if it's important to know who the caller is; don't use it if the questionnaire is supposed to be anonymous. Anything you type after this special symbol is treated as a comment and is not displayed.
- / <prompt string> Get a single line of text (up to 78 characters) from the caller. You can add a prompt string immediately after /; the prompt will be displayed exactly as you type it.
- +*n* <prompt string> Display a multiple-choice question, where *n* is the number of possible choices (up to 9). You can add a prompt string immediately after *n*; the prompt will be displayed exactly as you type it. The caller must answer with a single digit. If the caller answers with the highest numbered choice, you can add one of two options in the questionnaire file immediately after the +*n* line:

? <prompt string> Get a single line of text (up to 78 characters) from the caller as a further answer to the final choice of the multiple-choice question. This is useful if the last choice was "other" and you want further qualification.

! Terminate the questionnaire. This is useful if you ask a question like "Do you want to fill this out? (1) yes (2) no"; if the caller enters a 2, the questionnaire ends.

**? and ! are special:** The ? and ! options work only if (a) the immediately preceding question was multiple choice, and (b) the answer the caller gave was the highest numbered one possible. Note that you can use only one of these options per multiple-choice question.

A blank line anywhere in the questionnaire ends it.

Here's what the whole file might look like:

```
This is a private system. You can join if you meet certain
requirements. You'll have to fill out a brief questionnaire to
be considered.
*
+2 Do you want to fill out this questionnaire? (1) Yes (2) No
!
Are you now, or have you ever been, a politician?
+3 (1) Yes (2) No (3) Let me explain...
? This better be good:
Name all the people in our nation's capital whom you admire.
/Take up to 72 characters to answer:
Want to add any more comments?
/
/Please leave your phone number in case we accept you:
Thanks for your answers. We'll get back to you.
```

## Reading the Answer File

Here's what a typical answer file might look like:

```
* John Scribblemonger 14 Aug 87 23:12:53
  1: 1          [Caller answered Yes to "Want to fill this out?"
  2: 3          [Answered "Let me explain..."]
?  2: I volunteered for Dog Catcher.[Further answer to question 2]
  3: The ground's keepers at the Lincoln Memorial do a hell of a job.
  4: I believe I'd be a credit to this board and All It Stands For.
  5: (202) 456-1414
```

A line that starts with \* holds caller information.

A line that starts with ? is a further answer to the immediately previous multiple-choice question, and therefore has the same number as the previous answer.

A line that starts with spaces is the [first] answer to a question.

Answer numbers start at 1 for all callers, so you can tell where the next caller's answers begin.

## Three Kinds of Questionnaire

A system can actually have three separate questionnaires, two for new callers and one for callers who use the A)ns-Questionnaire command in the Main menu. What happens after a



caller fills out a questionnaire depends (among other things) on which questionnaire was answered.

The questionnaire for Anonymous-Questionnaire must be called MAINSECT.QES (as in MAIN SECTION QUESTIONNAIRE); its answer file is MAINSECT.ANS. Callers who finish filling out this optional questionnaire go back to the Main section and continue their business.

### **The Private System's Screening Questionnaire**

In private systems, people with no passwords (or with passwords not recognizable to Fido) get to fill out a questionnaire with the required name NOPWD.QES, assuming a questionnaire by that magic name exists. Answers go into a text file called NOPWD.ANS. (You create the questionnaire the same way you create MAINSECT.QES; the only difference is the name.) When a caller finishes filling out this questionnaire, s/he's logged off. After you screen an applicant's questionnaire, you telephone or write this new caller and provide a password. See "Setting Up a Private System", elsewhere in this chapter, for the details.

### **The Public or Semi-Private System's Questionnaire**

In public or semi-private systems, Fido will allow new callers to log into the system. Right after they choose a password, the text file NEWCLR2.BBS is displayed. However, if a questionnaire with the magic name NEWCLR.QES exists, Fido has the caller fill out this questionnaire instead; the caller's answers go into the file NEWCLR.ANS.

## Setting Up a Private System

A private system is one in which only callers who already have passwords can log on. You can let new callers fill out a questionnaire, let them leave a message for you, or you can just log them off (with or without your giving them the reason that they're being logged off). In this kind of system, you have to register each member manually with the SYSOP.EXE program (described in Chapter 2).

If you set the Private variable in FIDO.INI to Yes, the names of new callers aren't added to Fido's Caller list and new callers can't set up passwords for themselves. What happens next is up to you: You can give them a file of information to read, and after they've read it allow them to leave you a message; let them read the file and log them off without accepting a message from them; or let them fill out a questionnaire, after which you immediately log them off.

### Display a File of Information

If you've created a text file called NOPWD.BBS (and assuming there's no file in the directory called NOPWD.QES), its contents is displayed. In this file, you might explain that this is a private system, and you aren't accepting new callers; or you might explain the purpose of the system, and what a caller has to do to join.

After this file is displayed the caller can leave you a message — Fido issues an automatic G)oodbye command — assuming you've turned on the G)oodbye message prompt by using O=G for some message area in AREAS.INI.

### New Caller Questionnaire

If the questionnaire NOPWD.QES exists, Fido presents it to the new caller. This questionnaire lets you gather information on potential new callers, whom you then contact (or not) to issue passwords. (See "The Questionnaire" elsewhere in this chapter for details.) After a caller finishes filling out this questionnaire, s/he's logged off.

### InfoFile in Questionnaire Disguise

A questionnaire can contain text with no real questions. You can put the same text into the new caller questionnaire NOPWD.QES that you'd ordinarily put into NOPWD.BBS. The difference is that NOPWD.QES logs a caller off as soon as its display is completed, while NOPWD.BBS looks for the G)oodbye prompt. Using this last method lets you give information to new callers without having to deal with any questions from them, and still have the G)oodbye message prompt operating for system members.

**For the truly rude:** You can log a new caller off unceremoniously. If yours is a private system with an empty NOPWD.QES text file on it, a new caller is instantly dumped. This is a pretty nasty thing to do, but is good practice for those wishing to pursue a career in customer service.



## **Semi-Private Systems**

You can set up a semi-private system whereby new callers can log on, but have extremely limited access to commands. To do that, follow these steps:

1. Set the `Private` variable in `FIDO.INI` to `No`.
2. Set the `Default-Priv` variable in `FIDO.INI` to `0` or `1`.
3. Set correspondingly higher privilege levels for area and command access in `AREA.INI` and `COMMAND.INI`.
4. Put initial new caller information in `NEWCLR1.BBS`.  
For example, You can tell new callers whether or not you insist that they use their real names or not. (After the caller sees this file, they enter their name, address and choose a password.)
5. Put further new caller information in `NEWCLR2.BBS`, or have them fill out questionnaire `NEWCLR.QES`.  
If the questionnaire file `NEWCLR.QES` exists, the new caller fills it out. If it doesn't, then Fido just display the file `NEWCLR2.BBS`.
6. Enable the Goodbye message prompt to new users can leave you the information you need.  
You don't have to do this if they've filled out a questionnaire, of course; but new callers tend to be shy and might have some questions for you.

## Big Bang Time

After you've modified all the .INI, .BBS, .HLP, and BULLETIN files, you're ready to create the Fido universe. You do that by running SET-FIDO.EXE. Do this now:

```
SET-FIDO [CR]
```

Among other things, SET-FIDO.EXE takes the information you've supplied in all the text files talked about in this chapter, and converts that information into a form that Fido can understand. It also creates a host of system files and subdirectories that Fido needs to perform its daily operations. Some of those files create their own files, and after that it really gets messy. Luckily, you don't have to think about all this — it all happens automatically, and it's all for the best.

### Nothing Can Go Wrong (scratch) Go Wrong (scratch) Go...

SET-FIDO keeps you informed of what it's doing as it does it. If you've followed all the instructions in this chapter, the Fido universe should come into being without a hitch. If by chance some error does occur, SET-FIDO lets you know about it; the system has sagans of clear messages covering every error Fido's testers could come up with. These messages tell you what's wrong, and always give express or implied suggestions about how to clear up the problem.

**Rerun SET-FIDO:** When you correct an error by changing one of the .INI files, you must run SET-FIDO again, or Fido won't know you've made any changes. It's safe to run SET-FIDO as often as you want.

In the unlikely event that you get an error message you can't figure out, call Fido Software at the telephone number on the disk label. (Please have your Fido serial number handy; Fido Software only gives help to its registered licensees — among which are you, assuming you have an original Fido system disk and this manual, and you purchased your software directly from Fido Software.) If you purchased your software through IFNA, contact one of the IFNA Help nodes.

### Testing the System

After a few minutes, you'll see the MS-DOS prompt, and you're ready to test the system. The fewer changes you've made to the .INI files, the less you'll have to test. Basically, you should make sure that everything looks the way it should, that commands match default locks and privileges, and so on. You'll also want to make sure that your modem is working properly.

Here are some steps to take to test Fido. This isn't the only way to do it, but it works (mostly).

#### 1. Make sure your modem is talking to Fido.

To do this, you'll need to start Fido. (If you're running Fido from floppy disks, put the program disk in drive A: and put the data disk into drive B:.) Assuming you've given the name FIDO to the subdirectory that holds your system, get to the Fido subdirectory from your console by typing

```
CD \Fido [CR]
```

([CR] means to press the Enter or Return key.) Then start the Fido program by typing

```
FIDO [CR]
```



You should see the copyright notice and a message that says "Initializing modem." If all goes right, in a moment you'll see another message, "Waiting for call or event." When you see this, you know that Fido and your modem are talking to each other — which means that your modem is connected to Fido properly, which is 90% of the battle. (You don't know, however, whether your modem is working correctly; you'll test that in another step.)

## **2. Check the Events list.**

To check the Events list, type

?

(You don't need to press Enter here.) Fido displays a list of all the scheduled events, including the time in minutes until each event is scheduled to begin. (You'll also see a list of available commands, most of which are discussed in Chapter 2.) Fido also tells you if you have no events scheduled. Make sure what you see on the screen matches what you meant to type into EVENTS.INI. If something isn't right, shut down Fido by pressing CONTROL-C and look at EVENTS.INI in your word processor. After you've fixed the problem, you'll need to run SET-FIDO again. Or you might want to make a note of the problem and continue to test Fido, saving all your changes to do at once.

## **3. Check the initial sign-on files and messages.**

The goal here is to make sure that Fido greets callers the way that you want it to, and to make sure you've told Fido to say what you really mean to say. To do this, you need to enter Fido as if you were calling it through another computer. Fido has a Local mode that you can enter from the console. Type

L

You'll see the Fido sign-on, a display of the text file WELCOME1.BBS. Make sure it says what you want it to; everybody who calls will see it! Note down any changes you want to make, and go on to the next step.

## **4. Check the sign-on sequence for new callers.**

(You can skip this step if you're running a totally private system.) The new-caller sequence presents all the .BBS files (assuming you have a public board). Fido will prompt you through the process. Sign on with the name and password you really want to have. You'll get the default privilege level that you set in FIDO.INI. (In Chapter 2 you'll change your privilege level to 7, the level of the system operator — which, after all, is only fitting.) Check all the messages that come up, and note what changes you'll make later. They appear in this order (with the unchangeable messages in square brackets coming from Fido itself):

NEWCLR1.BBS  
[Fido's built-in address prompt]  
[Fido's built-in "Choose password" prompt]  
NEWCLR2.BBS  
[Fido's built-in password test]  
WELCOME2.BBS  
A quotation from Fido's QUOTE.BBS file  
MAIN MENU prompts from COMMANDS.INI

If you don't see a message that should be there, you've somehow left a file off the program disk. If you want the message to appear, you'll need to copy the proper text file(s) to the program disk, or create it (or them) with your word processor.

**5. Check the MAIN MENU help commands and any bulletins.**

To do that, type

? [CR]

The information in MAIN.HLP appears when you ask for help; the information in the BULLETIN.n files come up when you ask for them. Make sure that the text of the help is appropriate to the people who will be using your system. If you get a message saying that help isn't available, you've forgotten to copy that file to the program disk. The help files supplied with Fido are complete; but if you want to change them, see "How Help Works", earlier in this chapter. (Just before "How Help Works" is a list of all the .HLP files.)

**6. Check the Main Menu prompts and help files.**

Make sure that the commands that appear match the ones that you want to appear for someone with the default privilege level. This is especially important if you've made any changes, either to the command names or to the privilege levels associated with those commands in the COMMANDS.INI file.

**7. Check the Message section prompts and Help files.**

To do that, enter the message section and follow the procedures outlined in steps 5 and 6.

**8. Check the Message section Areas list.**

Make sure that the areas have the names and descriptions you want them to have. If something is amiss, note any changes you want to make; later, you can edit AREAS.INI.

**9. Check the Files section prompts and Help files.**

To do that, return to the Main Menu prompt; then enter the message section and follow the procedures outlined in steps 5 and 6.

**10. Check the Files section Areas list.**

Make sure that the areas have the names and descriptions you want them to have. If something is amiss, note any changes you want to make; later, you can edit AREAS.INI.

**11. Choose G)oodbye to make sure the sign-off prompt asks or doesn't ask for a message to the system operator, as you prefer.**

If there's no prompt for a message and you want there to be, you'll need to change AREAS.INI by adding an O=G option to one of the Area lines, or by creating an area just for Goodbye messages. Make a note and change it later.

**12. Sign on again as the system operator to check all the prompts and areas (including file upload and goodbye message areas) reserved for callers with the highest privilege level.**

Sign on with the name Studley Whackburn and use the password Tietack. Fido will admit you as the system operator. (Studley is the president of the American Tie Clasp Association, and one heck of a guy; in Chapter 2's "SYSOP.EXE" section, you'll depose Studley and install yourself as system operator — proving once again where nice guys finish.) Repeat steps 5 through 10, but you don't have to check the Help systems again.



**13. Quit Fido and modify files as necessary.**

**14. Run SET-FIDO again.**

**15. Repeat step 12 to check the places you changed.**

By now, everything should be in great shape. If you need to, note any final changes and make them as before.

**16. Call from an outside system to check your modem.**

Don't skip this extremely important step. Enlist the aid of a friend with a modem to call for you, if you want. The idea here to make sure that somebody can call from the outside. Read Appendix C, "Modem Basics", if something untoward happens.

**17. Get a friend to call who's never used Fido.**

Have your friend check all the messages, bulletins, and Help files to make sure they work. If they do, congratulate yourself. If they don't, rewrite the files and congratulate yourself for having the integrity to put a caller's needs over your own ego.

If you intend to use FidoNet, Fido's electronic mail component, you'll need to make sure that your mail system is working properly. See Chapter 3 for details.

## **Go For It!**

Congratulations! Assuming you've run your system to check everything out, you're ready to go on-line. Expect to make some changes over the first few days that your system is up as you learn more about Fido. You might need to edit your text files as you get feedback from your callers, or you might want to add or change some of your message and file areas as your board takes shape over time.

<p><b>Made any changes?</b> Remember to execute SET-FIDO again whenever you make any changes to any of the .INI files. If you don't, Fido won't know that you've made the changes.</p>
--

Be sure to read Chapter 2, "Running the BBS", to learn about Fido's day-to-day operations, and to see how to further customize your system. Read Chapter 3, "FidoNet," if you intend to send or receive FidoMail.

And welcome to System Operatorness.

## Chapter 2- Running the BBS

"Once you give up integrity, the rest is a piece of cake."

— J.R. Ewing

This chapter covers the day-to-day operations of Fido's electronic bulletin board system (BBS) from the point of view of the system operator (sometimes called the *sysop*). It assumes that you're already familiar with the way the BBS works from the caller's point of view. (If you're not, read Chapter 4, "Fido V12 Caller's Guide".) Details on the daily maintenance of the mail system are covered in Chapter 3, "FidoNet."

### The System Operator's Job

A system operator is anybody with a privilege level of 7, the highest level in Fido. A system operator has absolute power. (A BBS is a publishing medium, and the power of the press *a/ways* belongs to the person who owns it.) As system operator you decide the hours your board will run (if at all), who gets to use what sections of it, who gets to use what commands within sections, how long messages stay on the system, and even who can use the system.

As with most positions of power, being a system operator involves certain responsibilities. They fall into several natural categories. The first category of responsibility involves keeping the system going:

- Rid the system of outdated messages and files to free disk space and to make it easier for callers to scan
- Delete zero-length and other incomplete files
- Renumber messages to keep numbering from getting out of hand
- Update the log by dropping callers who haven't used the system recently
- Make sure your bulletins are current
- Assign keys and change privilege levels as appropriate ("appropriate" means whatever conforms to the philosophy of your board)
- Schedule ancillary events so as not to interfere with the main purpose of the board

The second category deals with the Law. As a publisher, you have to protect yourself from being hauled off to the Bastille:

- Purge the system of slanderous, libelous, and illegal messages (such as those that list stolen credit card numbers)
- Check uploaded files to make sure they're shareware or are in the public domain

The third category is the category of moral and ethical responsibility. You can put all the items in the legal category here, and add these:

- Ensure that downloadable files aren't malicious, that they won't do damage to people's systems when they run them (worms, trojan horses, etc.)
- Respond to questions left by callers at sign-off, or don't provide an area where callers can leave questions for you
- Don't read private mail
- When you purge a libelous, scandalous, or illegal message, leave a message for the offending caller telling why you deleted the message

The bottom line is that system operators have much the same powers and responsibilities of newspaper publishers and radio and TV broadcasters. This is especially true if the board is a



public one.

The rest of this chapter covers the first category's elements: the physical maintenance of the board. The law and the ethics are up to you.

### Console Commands

Fido provides a series of commands you can execute from the console. (In this case, the *console* is the terminal hooked to the computer that's running Fido.) You can enter Fido as if you were calling from an outside line (except that you don't go through the modem), you can perform some maintenance functions while Fido is between calls or scheduled events, and you can interact with a caller or work with a caller's records while s/he's on line.

You can also spy on a caller, observing each keystroke s/he makes, watching everything s/he does, every supple movement of the cursor, every sensual — oh. Sorry.

Here's a summary list of all the console commands. Commands marked with an asterisk (\*) have a more detailed explanation following the list:

#### While system is idle:

- |    |   |
|----|---|
| L  | Log into Fido in local mode as a caller. (A calling modem hears the phone "ringing" but nothing answers.) |
| I  | re-initialize the modem (in case you'd done something like turned the modem off by accident)              |
| ^C | Cancel Fido and to return to DOS  |
| R* | Reset the events pointer  |
| ?  | Show this list plus show scheduled events   |

#### When caller is on:

- |     |  |
|-----|--|
| C   | Chat with the caller   |
| Z   | Clear current caller's limits (accumulated download, time on system, accumulated time)                             |
| ^A* | Simultaneous keyboards; all console keystrokes act as if they were typed from current caller's keyboard thru modem |
| ^C  | If no caller is on, stops FIDO; exits to DOS   |
| ^X* | Terminates call; logs caller off   |
| ^T* | Disables modem after current caller logs off; notifies Sysop   |
| ^Z  | Terminates chat, paged conversation, simultaneous keyboards, and what ^T does                                      |
| ^O  | Turn off local screen output (speeds up multi-DOS programs)  |

<b>^N</b>	Turn on local screen output
<b>Spacebar</b>	Delivers "Sysop not available" message in response to a page
<b>?</b>	Gets description of all available commands

Here are the details:

- R:** This command resets the pointer that tells Fido what event lines from the EVENTS.INI file have already been executed. Essentially, this means that Fido can rerun all events that have already run (assuming that the current time is still within the event's WINDOW).
- ^A:** This is a great hand-holding tool for terrified new callers. You can walk the caller through a tutorial; what you type at the console acts upon Fido as if the caller were typing the commands on his keyboard.
- ^X** Use this command to kick somebody off Fido without letting them know that s/he's being kicked off. When you execute this command, Fido spits random characters that look like line noise onto the caller's terminal, then cuts the caller off. The ethics of using this command are questionable; the programmer has included it to allow you to test your moral fiber.
- ^T** This command tells Fido to stop accepting calls after the current caller logs off. When the caller hangs up, Fido beeps for up to ten minutes summoning you to the console. (A calling modem hears the phone "ringing" but nothing answers.) If you don't enter a command within those ten minutes, Fido will begin accepting calls again or will execute the next scheduled event.



## Command Line Switches

Fido has a few command line switches you can use to run Fido with certain options. (Previous Fido versions had sagans more of these switches; most of them have been integrated into FIDO.INI, described in detail in Chapter 1.) To use a command line switch, add the switch after the program name FIDO at the DOS prompt before you press Enter:

FIDO /2

Here's a description of the command line switches available in Fido Version 12:

**/L** Starts Fido in local mode from MS-DOS. You log on as if you were a caller, but you don't go through the modem and the EVENTS scheduler doesn't operate.

**/T** Same as /L.

**/1** Tells Fido to use COM1 instead of COM2. Use this to override the default setting of IO-PORT set in FIDO.INI.

**/2** Tells Fido to use COM2 instead of COM1. Use this to override the default setting of IO-PORT set in FIDO.INI.

**n/I** Sets task ID number for aftermarket multitaskers. This is for multitasking environments where several Fidos are running at once on the same machine, keeping separate logs but otherwise sharing message and other files. Giving a value greater than 0 to *n* forces Fido to recount messages before saving a new message. ( In a multitasking environment, two people on different Fidos might try to save different messages in the same area at the same time; if Fido doesn't recount messages just before saving, one message might have the same number as the other, thus wiping one of them out.) *n* — a number in the range 1 through 255 —is assigned to the *n* position in the files FIDOn.LOG and FIDONETn.LOG. (See Appendix B, "Files and their Functions.")

## Managing Uploaded Files

FILES.BBS is a text file that resides in each file area subdirectory. It holds the name and description of each uploaded file in that particular area, as provided by the caller who uploads the file. Each new entry is appended to the bottom of the list. The contents of this file gets formatted and displayed whenever a caller uses the Ffiles command.

FILES.BBS gets created in one of two ways:

- As system operator, you create it when you set up the board.
- Fido creates it (if it doesn't already exist) the first time a caller uploads a file to that particular area.

You can add file names directly to FILES.BBS by using the syntax `FILENAME.EXT comment:`

```
CALLER.DOC The compleat caller's guide to Fido
EDITOR.EXE A very decent text editor
ZIP.TXT All ZIP mail codes
IRANGATE.TXT Secrets the Hearings didn't uncover
```

(Be sure to include a space between the filename and the comment.) Don't format your entries to look pretty; Fido does that for you. Fido looks on the disk for the file. It puts up the file's name and file size, followed by the comment. If it can't find the file, Fido puts up the notice `MISSING` after the file's name:

```
Command (?=help): F
CALLER.DOC      58398 The compleat caller's guide to Fido
EDITOR.EXE      9130  A very decent text editor
ZIP.TXT         8031  All ZIP mail codes
IRANGATE.TXT MISSING Secrets the Hearings didn't uncover
```

## Adding Notes

You can add notes to FILES.BBS by starting a line with either a space or a dash. Fido displays the notes exactly as you've written them, so you can add "white space" where you want to make the display more attractive:

```
Command (?=help): F
```

```
  This section is reserved for text editors and text files of all
  types.
```

```
  Feel free to download as many as you want.
```

```
CALLER.DOC      58398 The compleat caller's guide to Fido
EDITOR.EXE      9130  A very decent text editor
ZIP.TXT         8031  All ZIP mail codes
IRANGATE.TXT MISSING Secrets the Hearings didn't uncover
```



## .Displaying File Creation Information

You can make Fido show the file creation time and date by adding %D to the start of a line. All files after %D will show the added information:

```
CALLER.DOC    20 Jul 87  1:29 58398 The compleat callers guide
EDITOR.EXE    4 Jul 87 17:41  9130 A very decent text editor.
ZIP.TXT       17 Jul 87 18:27  8031 All ZIP mail codes
IRANGATE.TXT                     MISSING Secrets the Hearings didn't uncover
```

## Using @ to Limit New File Display

In Chapter 1, you were urged to create a separate Upload subdirectory for each file area; the argument was that it's in your best interests to be sure all uploaded files are legal ones — that is, are in the public domain or are shareware. The trouble with that method is that it entails extra work: After you've checked a file, you have to move it to the download area, and you have to move the description line in the Upload subdirectory's FILES.BBS to the FILES.BBS in that area's Download subdirectory.

The @ character provides a quick-and-dirty kludge to let you keep uploaded files in the same area as downloaded files without callers knowing they're there. If you place the character @ at the start of a line in FILES.BBS, any new uploads listings below it won't appear when a caller with a privilege level less than 4 uses the F)iles command. Here's what FILES.BBS might look like:

```
...
ZIP.TXT All ZIP mail codes
IRANGATE.TXT Secrets the Hearings didn't uncover
@
WS.EXE Stolen copy of WhizzyStar.POWER TO THE PEOPLE!
```

And here's what a caller (privilege level 3 or lower) would see:

```
...
ZIP.TXT          8031 All ZIP mail codes
IRANGATE.TXT MISSING Secrets the Hearings didn't uncover
```

Later, after you've checked the new uploads, you can move the @ character to the bottom of the list again and the new name(s) will show up. Or, as in this case, you can remove the offending program before it gets out.

## Moving @ with the K)ill Command

The way that the `K)ill` command works provides an easy way to move `@` to the bottom of the `FILES.BBS` list. `K)ill` first deletes a file's name from `FILES.BBS`, and then tries to delete the file itself. Because it appears on the list, `@` is considered a file name by Fido. So if you tell Fido to `K)ill @`, it removes `@` from the list. You can then `U)pload` the "file" called `@`, thus inserting `@` at the bottom of the list in `FILES.BBS`.

**@ protection is shallow:** Realize that the `@` character doesn't stop anybody from downloading a file; it just hides the file's name. If somebody knows, for example, that `WS.EXE` is in a file area, s/he can download it whether or not the name shows up in a `F)iles` listing (or even if the file's name isn't in `FILES.BBS` at all).

**The file name `FILES.BBS` has magic properties:** only callers with privilege level 4 and higher can download or upload this file, regardless of any other access controls you might have. This prevents unauthorized peeking at files that are in the list following the `"@"` line; in otherwords, files that you haven't screened yet.



## SYSOP.EXE - Fido's Caller Clean-Up Utility

SYSOP.EXE is a simple list manager that maintains the caller database, CALLER.SYS. You use it to change records individually and collectively:

Command:

```

+-----+
| L)ocate      > Next      < Previous    B)eginning  E)nd        I)np)ut-New |
| N)ame        A)ddress     W Password   K)ey        $ Credit    P)rivilege   |
| F)ile-Area   M)sg-Area    S)amp        U)n-Stamp   O)ld-Stamp  ! Purge-Stamped |
| G)oodbye     C)heck-Recd ~ Cancel    * Copyright  ? Help!   |
| Fido/FidoNet SYSOP Utility - Copyright Tom Jennings 20 July 87 |
+-----+
|                                     COMMANDS                                     |
+-----+
| RECORD #1 |
| LAST CALLED: 31 Jul 87 19:55:27 |
| TIMES CALLED: 1 |
| NAME: Studley Whackburn |
| PASSWORD: Tietack |
| ADDRESS: Small South American Country |
| MSG AREA: 1 FILE AREA: 1 |
| UPLOADED: 0K DOWNLOADED: 0K |
| ACCUM. D/L 0K ACCUM. TIME: 57 min. |
| CREDIT: $0.00 DEBIT: $0.00 |
| PRIVILEGE: 7 |
| KEYS: |
+-----+
|                                     CALLER RECORD                                     |
+-----+

```

The file that SYSOP.EXE manages, CALLER.SYS, holds one record for each caller no matter how many times that caller logs on. A new record for each new caller is added to the bottom of the list.

SYSOP.EXE automatically saves the changes you make as you make them. Here's a description of all the SYSOP.EXE commands. Commands marked with an asterisk (\*) have a more detailed explanation following the list:

### Select record

B	Beginning of list (first record)
E	End of list
> or +	Next higher in list (closer to end)
< or -	Next lower in list
#	Specific record
L*	Locate specified name (substring)
^R	Repeat last L command

### Change selected record

N	Name (1 to 3 words)
I	Add new caller record to end of list
A	Address
W	Password
K*	Keys
\$*	Set (or add) credits and debits for FidoNet mail costs
P*	Change privilege level (0 - 4, 7 [7=SYSOP])
F	Set default File area (where first-time caller goes when s/he enters File section)
M	Set default Message area (where first-time caller goes when s/he enters Message section)

S	Stamp caller (usually for purging)
U	Unstamp
C	Check record integrity (enforces character limits; changes out-of-range-values to defaults; resets accumulated downloads field & time on system field; converts embedded control characters to @-signs)
~	Cancel changes to current record

#### Change records collectively

O*	Stamp Old callers
!*	Purge stamped callers

#### Special commands

esc	Abort command at current record (works for I, F)
*	Show copyright information
G	Goodbye; Exit to DOS, saving changes first

Here are the details:

- L:** The Locate command looks for a string of characters in the Name field that matches what you type. Sysop does a substring search; **mong** finds such names as Mongain, Among, and Scribblemonger. The search begins at the record following the one on the screen. Sysop searches until it finds a match and puts that record on the screen. If no match is found between the current record and the end of the list, Sysop wraps to the top of the list and continues. If it finds no match at all, Sysop tells you so; you end up with the record you started from still on the screen.
- K:** Use the Keys command to change the keys list. You can add up to 32 keys, numbered in the range 1 through 32; use a comma between keys. Type N (for NONE) to remove all keys. To remove a single key, type a minus sign, followed by the key number.
- \$:** Useful only if your system uses FidoNet, this command changes the amount of money a caller has in his account to pay for electronic mail. The number you type here is added to (as opposed to replaces) the current credit. To subtract money, use a minus number. To clear the field, type N. (FidoNet automatically deducts the appropriate amount from this field when the caller sends chargeable mail; see Chapter 3 for details.)
- P:** You used COMMANDS.INI and AREAS.INI to set privilege levels that determine who can use which commands and which message or file areas. Here, you use the P command to change somebody's privilege level to match. Changing somebody's privilege level has other consequences. The different privilege levels have — well — certain privileges (and in some cases restrictions) that go along with the rank. Levels 0 and 1 can't leave private messages, and have only half the usual time on system and Kbyte download limits. Level 3 has double the normal time and download limits. Level 4 has double time and download limits, the ability to use File Attach when sending FidoMail, the ability to kill any message or any file, and has the same access to FILES.BBS as does the system operator. Level 7 people need no keys, have no limits of any kind, and essentially have complete control of the system at all times; such people are gods (and should be treated as such).
- O:** Use this command to stamp the records of all callers who haven't called the system in a certain number of days. (Fido asks you to tell it what that number is.) All this command does is mark records; it doesn't do anything else to them. It's like doing a mass Stamp command. To unstamp a single record, use the U command; to unstamp all records, use the O command to stamp all callers who haven't called in some absurd number of days —1000 for example.



**!:** This command removes all records that you stamped using the S or O commands. First it makes a backup copy of CALLER.SYS called CALLER.BAK; then it moves all stamped records from CALLER.SYS to a file called CALLER.OLD. To undo such a purge, rename CALLER.BAK as CALLER.SYS or append CALLER.OLD to CALLER.SYS. This command won't remove callers with a privilege level of 7 or who have any FidoNet credit left; to remove such records, first set the credit to N and lower the privilege level to 4 or below.

## **Small South-American Country Time**

It's now time to depose Studley Whackburn and install yourself as System Operator. Here's how to do it:

- 1. Use SYSOP.EXE to get to Record 1.**  
It comes up automatically.
- 2. Use P to set the privilege level to 0.**  
You have to do this before you can delete his record. (The real first step is to rid Whackburn of his power — no American advisors required.)
- 3. Use the S command to Stamp him.**  
This step marks the record for later action.
- 4. Use the ! command to purge the stamped record.**  
The system whirs a bit as Whackburn's record goes away. Your own record, formerly the second one, becomes first.
- 5. Use P to set the privilege level to 7.**  
This installs you in the position of ultimate power (which is, after all, your birthright).

## MSGMGR.EXE - Automated Cleanup

The Message Manager purges the system of out-of-date messages, and removes from the caller list people who haven't called in a certain number of days. The Message Manager is made up of two parts: MSGMGR.INI is a text control file that holds the commands you want to execute; MSGMGR.EXE compiles and carries out the commands in MSGMGR.INI.

There are but three commands: Age, Purge, and RENUM.

<b>AGE <i>number of days</i></b>	Sets the number of days for the next PURGE command(s). AGE is in effect until you enter another AGE command.
<b>PURGE <i>area list</i></b>	Remove from the areas listed any messages older than the number of days given in the most recent AGE command. Separate entries in the list with commas or spaces.
<b>PURGE CALLERS</b>	Remove from CALLERS.SYS any callers who haven't called the system in more than the number of days given in AGE. The names of the purged callers are moved into a file called CALLERS.OLD (in case you change your mind).
<b>RENUM <i>area list</i></b>	Renumber the listed areas.

You must include either a PURGE or a RENUM command in the file; if you don't, nothing happens. The default age is 90 days.

You can use the modifiers ALL and NOT in appropriate places with message area lists:

<b>PURGE ALL</b>	Purge all areas of messages older than the most recent AGE.
<b>PURGE ALL NOT 3,7</b>	Purge all areas except areas 3 and 7 of messages older than the most recent AGE.
<b>RENUM ALL NOT 1,5,9</b>	Renumber the messages in all areas except 3,5, and 9

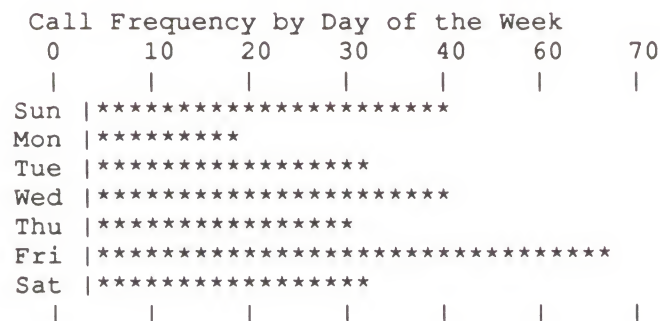
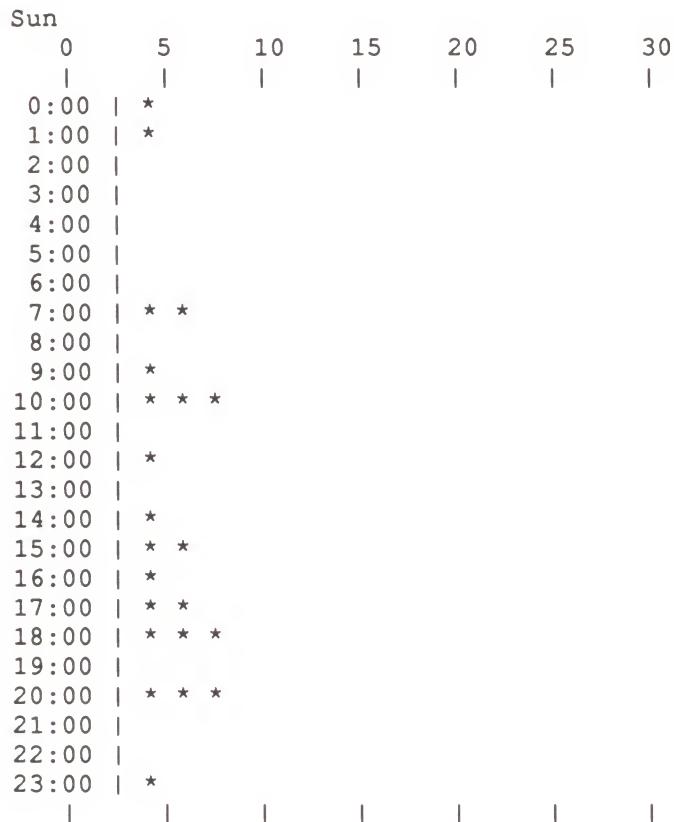
Fido always purges an area before it rennumbers it, no matter in what order you list the commands. The order in which you give the AGE command, however, is significant.



## TIMELOG.EXE - Fido's Call Counter

FIDO keeps track of the number of calls coming into Fido in a file called TIMELOG.BBS. The program TIMELOG.EXE uses the information in TIMELOG.BBS to create a bar graph showing how many calls come into Fido each hour. Over several weeks, you can tell what hours are consistently busy so that you don't schedule maintenance events in those hours.

The program produces two types of charts: one shows the frequency of calls on an hourly basis per day; the other shows the call frequency by day of the week. Here's a sample of the charts it creates:



## FIDO.LOG - Fido's Caller and Event Log

Fido keeps a running log called FIDO.LOG (logically enough) of what happens on the system. The file. When somebody calls, it keeps track of the caller's name, date and time called, baud rate used, the names of any files uploaded or downloaded, and a list of any errors or unusual events. FIDO.LOG is the best way to keep track of system activity.

Here's a typical log record. Note the dashed line that separates one log entry from another, and the unique character markers indicating the different kinds of events. The final line of a caller event shows the number of times this caller has been on, plus the time s/he spent on the system during this call:

```
-----
+ Bill E on at 08 Jul 87  13:00:44, 1200 baud
! Wrong pwd: Sissy
! Wrong pwd: Punk
+ 2 times, 2 min
-----
+ Matt Capps on at 09 Jul 87  12:36:26, 300 baud
! Yell, NA
+ 1 times, 7 min
-----
+ Tom Jennings on at 09 Jul 87  12:58:59, test mode
+ 53 times, 10 min
-----
Incoming Mail at 13 Jul 87  16:55:27
* Receiving Mail Packet
* Received packet
* Received 1 files
* Packet from Fido 0:107/9249
  Files: WARNT.ARC
-----
Mail done at 13 Jul 87  16:56:16
+ 176 times, 3 min
-----
+ Tom Jennings on at 14 Jul 87  16:59:11, test mode
= UL to scotfile\ : AREAS.INI
COMMANDS.INI
FIDO.INI
MSGMGR.INI
EVENTS.INI
+ 176 times, 5 min
-----
+ Scot Kamins on at 15 Jul 87  09:29:25, 1200 baud
! cant find xfertype.hlp
! DL scotfile\events doesn't exist
= DL events.ini
  1 OK
+ 15 times, 6 min
```

Because these files get large, you might want to copy or print them, and then delete the originals, on a regular basis. (A good way to do that is to create an external event using EVENTS.INI, and add the appropriate code to a batch file.)



## Chapter 3- FidoNet

FidoNet is a highly sophisticated electronic mail system. It links together FidoNet-protocol bulletin board systems (BBS) through telephone lines in such a way that anyone using any FidoNet-protocol BBS can exchange messages and files with any other FidoNet BBS. Using Fido's automatic scheduling system, any number of BBS's can exchange mail unattended at any convenient hour; or each BBS can act on its own, sending its mail independently. FidOs with version number 12 and higher (including the system this manual describes) are always ready to receive mail.

FidoNet currently supports up to 65,000 completely independent public or private stand-alone systems (nodes). As of this writing, nets are operating with as few as two and as many as 2000 separate nodes.

### Setting Up Your Own Network

You can either set up your own network or join an existing one. Setting up a network can take a lot of work, especially if each Fido is controlled by a separate individual or organization. It entails a great deal of cooperation. Here are some of the issues to consider:

- Do members belonging to another net (IFNA, for example) affect our net?
- Should there be a host for our network to handle mail interactions with other networks?
- Do you need a network administrator to handle the assignment of node numbers and other administrative details?
- Will you have a regularly scheduled hour during which all mail is exchanged?
- How will costs be handled? Will each node be responsible for its own mail expenses?

## Choosing a Network Topology

The way that you set up your network and mail schedule depends on (among other things) the number of nodes you're dealing with, the geographic distribution of those nodes, the amount of FidoMail the nodes need to exchange, how quickly and how often you want to exchange FidoMail, and the amount of money available to pay for telephone charges.

For example, you might set up an amorphous network with a near real-time FidoMail delivery schedule. In this setup all nodes have direct access to all other nodes: to send a piece of FidoMail, a node in Chicago for example places a call to a node in London — perhaps within moments of the FidoMail being posted in Chicago. The advantage of this method is that it's simple and quick; the disadvantages are that the telephone costs can be extremely high (unless all the nodes are in the same area, or you have direct access to a private telephone system with its own satellite), and if the node to receive mail is a popular one you might have trouble getting through the human and electronic traffic. All things being equal, this setup works most efficiently and economically in a relatively small network (say, 50 nodes or fewer) in the same geographic area.

A more complex and larger network might involve a two-layer topology with a fixed schedule of FidoMail delivery. All nodes in Kansas, for example, might send all their out-of-state mail to a central node (called a host) in Topeka at some predetermined time — most economically in the middle of the night when telephone costs are cheapest. After the host node receives all this mail, it sorts the mail by state and nation, and forwards it to the host in the appropriate state or nation. The receiving host then forwards the mail it receives to the appropriate local nodes.

The advantage here is a two-way cost savings: first, telephone calls in the middle of the night are far cheaper than calls made during the day; and second, you can send several messages in the same minute in the same call. (Most systems charge by the minute, so that a 10-second call costs as much as a full minute; in one minute you can send a thousand 7-character words by modem — 7 characters is an average word length in the English language — which can easily account for 10 separate FidoMail messages at 1200 bits per second.) Additionally, such cooperative calling means that far fewer callers are trying to get through to a particular node, and middle-of-the-night calling virtually eliminates human competition for connect-time. The disadvantage in this system, strictly followed, is that a piece of mail sent from one node might not get to another node as quickly as you might want; and if a host system becomes inoperative for some reason, lots of mail might be delayed for one or more days.

The actual system you set up depends on your needs. Fido is flexible enough to handle up to a three-layered topology based on any schedule you care to set. In its most extreme, this means that you can send mail to a regional host (who gets mail from all over Kansas) who then sorts and sends its mail as appropriate to a zone host (who gets mail from all of the Central United States) who then sorts and passes the mail as appropriate to other zone hosts (Western United States, London, Paris, Moscow, etc.) who then distribute the mail as appropriate down to their own regional hosts, and so on. You can set FidoMail schedules such that Fido sends mail as often or as seldom as you like.



## Joining the IFNA Network

The largest currently operating FidoNet-based network is IFNA, the International FidoNet Association. This public independent network (set up with the cooperation of, but in no way affiliated with, Fido Software) had over 2000 nodes all over the world as of August, 1987. Unless you're a private organization with special needs, IFNA is probably your best bet for sending and receiving FidoMail. For information, contact:

IFNA  
Box 41143  
St. Louis, MO 63141  
(314) 576-4067 [Voice]

### Nodes, Nets, Regions, and Zones

When you join IFNA or any other network, you'll be assigned three special numbers. These numbers are meant to identify your BBS uniquely, and represent your Fido's address. Other BBS's will use this address to get FidoMail to your system. For example, a typical address is 1:125/111. (The colon and backslash serve as special delimiters.) When you get these numbers, you'll need to type them into the proper place in the file Fido.INI; then you'll need to run SET-Fido again so that your system realizes you've made these changes.

**Don't assign yourself numbers:** Unless you're the person designated within your network to assign zone, net, and node numbers, don't assign numbers to yourself. If you assign yourself numbers arbitrarily within a network and then start to exchange FidoMail, you're likely to cause incredible confusion and cause much mail to be misdirected or even lost. Networks are cooperative entities, and the rules by which networks work are decidedly not random, arbitrary, or capricious.

The first number in the address (1 in the example) is the *zone* number; it represents some vast geographic area, usually a country — in this case the United States.

The second number (125) represents your *net*, the affiliation of Fidos in your specific geographic area — the San Francisco Bay area in the example given. All members of a given net are usually in the same telephone area code, and tend to send mail destined for outside the net's area to the same *host node* who then forwards the mail appropriately to nodes within its own net. (Conversely, net members usually route all FidoMail going to Fidos outside their immediate net through their host for cost and time efficiencies.) If you live in a part of a country that has few Fidos, you'll receive a *region* number instead of a net number; it has the same purpose as a net number, except that members of a region tend to have different area codes and live farther apart from each other.

The third number (111) is your node number. You're the only Fido in your net or region who has this number. Somebody in another net might have that same node number, just as somebody in a different telephone area code might have same seven-digit telephone number. But it's the net number that distinguishes your number.

## How Mail Gets Sent

A caller sends electronic mail by leaving a message and/or file in the FidoMail section of the BBS. (This, of course, implies that as system operator you've set up a message and/or file section for FidoNet using the `O=FidoNet` option in the `AREA.INI` text file, described in Chapter 1.) Fido prompts both for the name of the person to receive the FidoMail and for the BBS address (described above) where s/he will look for the FidoMail. For a complete description of how to leave FidoMail, see "Leaving FidoMail" in Chapter 4.

### FidoNet Events

Fido sends FidoMail when it's time for an FidoMail event to take place. You tell Fido when to run such an event by typing FidoNet event lines into the `EVENTS.INI` text file described in Chapter 1. To refresh your memory, the event line says what day or days to run FidoNet, at what hour, for how much time, and using what FidoNet route file (described in the next section). A typical FidoNet event line looks like this:

```
ALL 2:00 60 FidoNet A
```

When it's time for FidoMail, FidoNet determines what FidoMail to send and where to send it. It creates packets of FidoMail by reading the addresses of all the FidoMail and bundling together everything going to the same address. (If there were six pieces of FidoMail going from your board in Boston to some board in Cleveland, it would be inefficient and expensive to make six separate long-distance calls when one would do.) In the absence of other instructions, FidoNet makes telephone calls and delivers its FidoMail; then it hangs up. Normally it waits for sixty seconds before making its next call (if it has one to make) to see if any other Fido system wants to deliver any FidoMail to it. (FidoNet also dials out immediately after receiving an incoming call.) This process continues until FidoNet delivers all its FidoMail, or until the window described in the `EVENTS.INI` events line expires.

### Route Files: FidoNet's Message Forwarding System

Route files control where and how to send mail. Actually, FidoNet can handle all its mail automatically; if your event line ends in the phrase `FidoNet A`, FidoNet sends the FidoMail in the most straightforward way possible. You use route files to route (get it?) FidoMail through a different path.

**The normal course:** Unless you tell it differently, your board sends FidoMail to the host of the net of your FidoMail's destination. For example, you have a piece of mail going to Revere, Massachusetts (a member of the Massachusetts net). Your outgoing FidoMail gets sent to the Massachusetts net host. The Massachusetts net host then forwards the mail it has for the Revere Fido. There are two exceptions to this through-the-destination's-host rule: FidoMail within your own net and FidoMail with files attached are sent directly to their destinations without going through the host.



A route file is a text file you construct with your word processor. A route file has the name `ROUTE.letter`, where *letter* is a letter from A through W. Like the .INI files, a route file is made up of commands and parameters: You construct the file, make sure it's in the Fido subdirectory on your working disk, and then run SET-Fido to let Fido know you've made some changes. When you construct a FidoNet event line in EVENTS.INI, you enter the event as FidoNet *letter* where *letter* is the route file you want FidoNet to use. You can construct up to 23 different route files.

**ROUTE.A is magical:** Historically so many people were using a route file that consisted of the single command `SEND-TO ALL` — instructing FidoNet to send mail as described in the box called **The normal course**, above — that it seemed only fair to build it into Fido. You can have an event line that ends in the phrase `FidoNet A` without actually having a file called `ROUTE.A`, and FidoNet will automatically do `SEND-TO ALL`. You can construct a `ROUTE.A` file if you want; just realize that FidoNet assumes that the first line in that file is `SEND-TO ALL`, whether you type it or not.

## Route File Commands

The order of commands in a route file is important: Where two commands contradict each other, the later command has precedence. For example, "Send to all" followed by "send to not all" sends no mail at all. (FidoNet reads and interprets the whole file before it packets and sends any mail.) Where the syntax allows a list, separate all items in the list with commas, spaces, tabs, or new lines.

Commands can take more than one lines. These examples are equivalent:

```
Route-To 99 1,2,3,4,78,79 101
```

```
Route-To 99
        1,2,3,4,78,79 101
```

Here's a description of all the route file command language keywords with their proper syntax:

`SEND-TO <list>` Send mail only to nodes in the list. This command must appear in your route file (except for the special `ROUTE.A`) or Fido will send no FidoMail. If `<list>` is the word `ALL`, Fido allows FidoMail to be sent to all nodes anywhere.

`ROUTE-TO <destina.>,<list>` Routes mail through `<destination>` for all the nodes in `<list>`. `<destination>` must have the command `ACCEPT-FROM <list>` in its operating route file in order for it to pass on the FidoMail sent to it. If `<list>` is the word `ALL`, Fido routes all its FidoMail through `<destination>`.

ACCEPT-FROM <list>

Forward FidoMail received only from the nodes on the list that follows. You can also restrict by saying ACCEPT-FROM ALL NOT <list of nodes> or ACCEPT-FROM <list of nodes> NOT <list of nodes>. If the board doesn't have an ACCEPT-FROM list, or if it doesn't have a specific address in its <list> parameter, the FidoMail is not sent on and is likely to be trashed (depending on the propensities of the system operator). If <list> is the word ALL, Fido will forward all mail with a forwarding address.

PICKUP <list>

Makes your Fido ask for mail addressed to it when it calls the listed nodes to deliver mail. If <list> is the word ALL, Fido picks up its own mail from every node it calls.

POLL <list>

Makes Fido call the listed nodes and ask for mail addressed to it. If <list> is the word ALL, Fido will call every node in the node list. (See "Node List", below.) It's not a good idea to use POLL ALL if your node list is provided by IFNA; the IFNA list has over 1800 nodes in it, covering all 50 states and at least 17 countries.

NO-ROUTE <list>

Makes Fido ignore any routing established for the listed nodes. Usually used with HOLD for later PICKUP. If not used with HOLD, Fido will send NO-ROUTE mail direct to the listed nodes (assuming there's any mail to send).

HOLD <list>

Stops Fido from sending mail to the listed nodes (although the mail is packeted for possible pick-up).

ZONE <num>

Changes preset zone number for further routing commands so that nodes in <list> need only include net and node numbers.

NET <num>

Changes preset net number for further routing commands so that nodes in <list> need only include zone and node numbers.

SEND-ONLY

Tells Fido to send mail as fast as possible, without the standard 60-second pause for incoming calls. After all mail is sent, Fido will accept mail until the end of the Window as set for this FidoNet event line in EVENTS.INI.

RECEIVE-ONLY

Tells Fido to receive mail only, and to send none. Waits for other systems to PICKUP or POLL for mail.

EXTERNAL-MAIL

Stops Fido from deleting packets and file lists at the end of a schedule. This is a special purpose command to allow moving mail outside of the FidoNet universe. Try not to think about it.



## Substitute <list> Words

<command> ALL	Tells Fido to make <command> apply to all nodes in the node list. (See "Node List," below.)
<command> MYNET	Tells Fido to restrict <command> just to the nodes in its own net. Not effected by NET command.
<command> THISNET	Tells Fido to restrict <command> just to the nodes in the current net (either default or set by NET).
<command> HOSTS	Tells Fido to make <command> apply to all hosts (and only to hosts) in the node list. A host node (and only a host node) has the node number 0. (See "Node List," below.)

## Negation

NOT <command>	Negates <command> or <list> ; makes <command> or <list> have the opposite effect. NOT stays in effect until the next statement. Thus the command line SEND-TO ALL NOT 125/4, 156/5 sends mail to everybody except 124/4 and 156/5.
---------------	--

## Node List

The node list is the master list of all nodes in the network. This master list resides in a text file called NODELIST.BBS. You must have a node list in your FIDO subdirectory for FidoNet to be able to work. (If you got the node list through IFNA, it's called NODELIST.*nnn*, where *nnn* is a three-digit number that defines the day of the year this node list was prepared. Be sure to run the proper IFNA-supplied programs before you use it.)

Each entry in the node list has the following form, with elements separated by a tab or space character:

Key	Number	Cost	Baud	Name	Phone	Location	Comment
-----	--------	------	------	------	-------	----------	---------

The following list defines what each element in a node list entry means:

Key	Tells what this line defines: If blank, defines a simple node in a net; ZONE defines ZONE host; HOST defines NET host; REGION defines a logical, as opposed to physical, net (FidoMail goes direct, as opposed to through a host); NODELIST identifies this as the node list itself, and everything after it in the line is ignored.
-----	--

Number	Node, net, or zone number (depending on the Key word). This number cannot be zero. If the keyword is NODELIST, then this number can be anything.
--------	--

Cost	How much you want to charge a caller, in cents per message, to send a message to this board from your location. When this cost is zero, it defines a call you don't charge for (usually a local call). If this number is greater than zero, the caller must have enough credit in his caller record to pay for the call; otherwise FidoNet won't send the message. As system operator, you change somebody's credit using the SYSOP.EXE utility described in Chapter 2.
------	---

Baud	The speed Fido will set your modem to when it's time to call this board. The number is always a multiple of 300, up to 9600. Fido won't set your modem to a number greater than it can handle.
------	--

Name	This board's name, up to 25 characters (all one word — underscore used for spaces).
------	---

Phone	This board's phone number, up to 39 characters. You can't use spaces, but you can use any of the following:
-------	---

Digits	1234567890
Special	# *
Ignored	( ) -
Two-second delay	.
Wait for tone	?
Wait for quiet	@
TouchTone™ (default)	T
Pulse	P

If your modem doesn't support a particular function, Fido ignores it or converts it to the closest matching function (usually a one-second pause).

Location	Where this board lives (all one word — underscore used for spaces)
----------	--

Comment	Whatever you want (all one word — underscore used for spaces)
---------	---



## What the List Looks Like

Here's what a typical node list might look like:

```
HOST 125    25    1200    SF_BAYNET      (415)-555-1234    SF_CA    24_hrs
      111    25    1200    FIDO_SOFTWARE  (415)-555-4321    SF_CA    Jennings
      123    25     300    RAD_SKATEBOARDS (408)-555-3758    Rampo_CA
HOST 165    35    2400    AMERICAN_TIE_CLASP (617)-555-3456    Lowell_MA
      906    35    1200    RANKNET        (617)-555-2345    Billerica_MA
...

```

The first node listed is the host; the number in the second column of the host-defining line says what net this is (net 125). A host's node number is always 0 — so the address of SF BAYNET is 1:125/0 (Zone 1, Net 125, Node 0). The second node listed is node 111; its net is 125, because the net of the most recently-listed host was 125. All nodes after a HOST line have the same net number as the host.

The net number changes at the next HOST line: SCOTO\_GATE is the host of net 165; its address is 1:165/0. RANKNET is node 906 in that net; its address is 1:165/906. And so it goes.

## Compiling the Nodelist

Once you've set up the nodelist, you must convert it to a form that FidoNet can use. You run the program MAKELIST.EXE, which converts NODELIST.BBS (the nodelist as text file) to NODELIST.SYS (the nodelist in a compiled form). FidoNet will handle the rest.

**Don't erase any files:** MAKELIST.EXE and Fido create several other files that make FidoNet work (NODELIST.SYS, NODELIST.IDX, NODELIST.NMP, and NODELIST.NTC). Don't erase these files or FidoNet won't work.

## Chapter 4- Fido V12 Caller's Guide

### Fido's Structure

Fido is divided into three major sections — Main, Messages, and Files. The Messages and Files sections can in turn have up to 200 separate areas each, each area covering a separate topic. (Most FidOs have fewer than 10.) Which of these sections and areas you can use, and the commands you can use in each, depends on your privilege level. The system operator determines what a caller's privilege level is.

The Message section holds messages that a caller writes to another person or to anybody who cares to read it. A caller can make a message private so that only the message writer and the person specified (and the system operator) can read it. The system operator determines the length of messages that callers can leave. Different types of messages usually go into different message areas.

The Files section holds files that a caller can download. These files come from the system operator and from callers who use special Fido commands to upload them. The system operator checks these files to make sure that they're either in the public domain or that the file's copyright holder has given permission for their distribution. Like messages, different types of files usually go into different file areas.

Many boards have a special electronic mail (FidoMail) section. In this section a caller can leave a message and/or a file for anyone who calls into a Fido board anywhere in the world. As of this writing, over 2000 Fido boards with FidoMail in operation are operating in all 50 states and 17 countries.

You'll read more later about entering and editing messages, uploading and downloading files, and sending and getting FidoMail.



## Calling Fido

When you call a Fido board, there's a process you have to go through before you can start reading or entering messages or files. It's called the log-on procedure (because you electronically sign your name in the log book). Except for the first time you call, it only takes a few seconds. Here are the steps you take to call and log on:

**1. Make sure your modem is turned on.**

Of course, it has to be attached to your computer properly, and you must have communications software running that works well with your computer and your modem.

**2. Dial the Fido's number.**

How you do this depends on your communication software; see the manual that came with that software for details. Assuming the Fido's number isn't busy and that the Fido you're calling is up and running, after a few seconds the word `CONNECT` (or something similar) will appear on your screen.

**3. Wait for a few seconds.**

This gives Fido a chance to check and match your modem's baud rate — the speed your modem is sending and receiving information. If you get no response after waiting five seconds or so, press Enter and spacebar two or three times. ( Older FidOs with version numbers less than 12 can't tell the baud rate of a modem automatically.)

**4. Read what Fido puts on your screen and follow the directions.**

You'll see the name of the board, and probably some introductory information the system operator wants you to read. Then Fido asks for your name.

**5. Type your first name and press Enter.**

(You can use Return instead of Enter.) Fido then asks for a last name. See the comments after this list in the section called "About Names."

**6. Type your last name and press Enter.**

(You don't have to type a last name if you don't want to; but if you don't type it the first time you call this Fido, don't ever type it when you call.) Next, Fido checks to make sure you've typed your name correctly

**7. Confirm that your name is typed correctly by pressing Y and then Enter.**

(Pressing just Enter without typing anything works, too; wherever Fido gives you a choice, pressing Enter is the same as pressing the capitalized choice plus Enter.) If you've made a mistake or somehow Fido got it wrong, press N and then Enter; then retype the information. Fido checks to see if it knows who you are. If it does, it asks for your password. If not, something else happens; skip to the section called "First time Log-ons."

**8. Type your password and press Enter.**

Fido makes sure you are who you say you are. If everything is OK, your log-on is complete. Fido might show you some notices; usually you can enter Control-C (or Control-K) to skip them. At any rate, Fido will soon present you with a deathless quotation that you can ponder as the Main Menu scrolls onto the screen.

## About Names

Some system operators want you to use your real name; others accept "handles" — aliases you use instead of your real name to reflect some fantasy or role appropriate to the kind of Fido you're calling ("The Hacker," "Top Whip", or "Creampuff" to name some typical handles). Custom dictates what's acceptable and what isn't on a particular Fido. In some cases, a system operator asks for your real name in a questionnaire and then allows you to use a handle everywhere your name would ordinarily appear on the board. In any case, you can sign on with whatever name you want; later, from within Fido itself, you can change your name and password.

Fido requires that you give a first name; whether you give a middle or last name is up to you. To skip giving a last name, just press Enter when Fido asks for it without typing anything else. You need to be consistent, however; if you sign on the first time with a first and last name, you must give both each time you call — otherwise, Fido will think you're somebody else. Also, if you give just a first name the first time you call, you can never give a last name when you log on. Again, you can change all this from within Fido (see C)hange command, below).

## First Time Log-Ons

Logging on for the first time takes a little longer (and sometimes a lot longer) than usual. You'll be asked to choose a password so that only you (and the system operator) will be able to read your private messages and so that nobody else can use up your time. (Many Fidos allow you only so much time on the system per day; some Fidos charge you for the time you use.) Some Fidos also ask you to fill out an electronic questionnaire the first time you call.

A few Fidos are semi-private or private systems; you have to be approved by the system operator before you can use most of the system's features. Some Fidos won't let you on at all unless you belong to a particular company or club, or unless the system operator has already enrolled you on the caller list that Fido keeps.

Whatever the procedure is, the Fido you're calling will let you know what's up and will prompt you for the information it needs. At any rate, you only have to go through this procedure the first time you call a particular Fido.

**Signing up - again!** All Fidos keep track of (among other things) the date that you call. Most Fidos purge your name from the caller list if you don't call again for so many days, usually 30. So you might find a Fido that you infrequently call forgetting who you are. When this happens, you'll have to go through the first time log-on procedure again.

## Choosing a Password

Your password can be up to 15 characters long. It must start with a letter, and it can't contain any spaces. It should be something that's meaningful to you and to no one else, or it should be completely meaningless. The idea is to come up with something that nobody else can use to rip off your messages or Fido time. Some people like to change their passwords at least once a month. (See "C)hange".)



## The Main Menu

Fido is a menu-driven system. This means that you choose commands from menus to decide what happens next. (Advanced callers can shut the menus off and type commands directly; see "Help Levels", for the details.) To choose a menu command, you usually type a letter and then press Enter. Here's what the menu in the Main section, the menu you see first, looks like:

```
MAIN Section:
M)sg-Section F)ile-Section B)ulletins
S)tatistics C)hange P)age-Operator
L)ist-Callers A)ns-Questionnaire V)ersion
O)utside E)xitDOS G)oodbye
Command (?=help):
```

You might not see all these commands if your privilege level isn't high enough. (In fact, it's extremely unlikely that you'll see either O)utside or E)xitDOS). Your system operator decides what commands you can see; if you can see it, you can use it.

**Privileges and such:** The system operator controls who can use which sections and, within sections, which commands. Different system operators have different rules to determine who gets to use what. System operators generally make it clear what you have to do in order to get to use these "higher order" commands. If you're curious, leave a message for the system operator asking about it.

Here's what each command does, in brief. The ones that require further explanation have their own detailed sections later on.

M — Message section. This brings you to the message area you last looked at on your most recent call. If this is the first time you've used the M command, you go to whatever message area the system operator sets for first-time callers. See "Message Section".

F — Files section. This brings you to the file area you last looked at on your most recent call. If this is the first time you've used the F command, you go to whatever file area the system operator sets for first-time callers. See "File Section".

B — Bulletins. Gives you a numbered list of bulletins you can look at. To see a particular bulletin, type its number and press Enter. The following example comes from the Ch@os board in San Francisco:

```
--- Bulletins ---
1) The current "New caller" welcome 7 July 87
2) THE INTENT OF THIS BBS
3) PRIVACY & CENSORSHIP
4) General notes on Bulletin Boards in the "real world"
Bulletin #:
```

S — Statistics. Lets you know how often you've called, how much time you've got left on this call, and other information. This comes up:

```
28 Jul 87 21:12:51
Total Calls:      Total    Limit    Remaining
Your Calls:      264
Connect Time:    41
                  1      60      59
```

The credit line lets you know how much credit you have in your account for using FidoNet, a special feature of Fido systems that lets you send electronic mail to and receive electronic mail from any of over 2000 Fido systems throughout the world. See "Sending and Receiving FidoMail", later in this chapter, for the details.

You're limited in the amount of time you can spend on the system, both per call and per day. Some systems are extremely busy, and they might limit you to less than an hour a day.

- C — Change your setup. Lets you change your name, password, help level, and system stuff (screen width and length, whether your modem needs nulls, and so on):

```

---- Personal Info ----
Name           : John Scribblemonger
Address        : Lake Woebegone Minnesota
Password       : Scribe
Help Level     : All

----- Your System -----
Number of Columns: 80
Number of Lines  : 24
"[More[Y,n]"    : ON
Tab Expansion    : ON
Filler Nulls    : 0

N)ame A)ddress P)assword H)elp-Level W)idth
L)ength M)ore T)abs F)iller-Nulls
Command (?=help):

```

See "Change Caller Settings" for details.

- L — List callers. This command shows you the names of all people currently on the caller list, the last date and time they were on, and where they're from. You can also hunt for an individual caller's name — a great way to see if you can leave mail for a friend, or to see the last time that somebody was on:

```

Name or part of name to match (CR = all):

Scott Dean      28 Aug 87  22:12:35  Fresno Ca
Jack Friday     06 Sep 87  09:40:53  Kathmandu Nepal
Al Taddeo       25 Aug 87  23:40:28  Burbank Ca
Ed Johnson      08 Sep 87  20:25:05  San Francisco Ca
Rob Taylor      19 Sep 87  20:41:57  Fresno Ca
Jeff McNeal     03 Sep 87  22:38:49  Pittsburg Ca

```

- P — Page the system operator. Sometimes the system operator is hanging around the computer room. Entering this command rings for the system operator. If s/he's around and feeling communicative, this will bring hir to the console.
- A — Answer the questionnaire. Some Fido systems have questionnaires on various subjects. Try this command; if there's no questionnaire on this system, Fido tells you so.
- V — Version number. This tells you what version of Fido this system is using. Many versions of Fido exist (over a dozen at the time this manual was written); the number of the version you're calling can be useful if you're a veteran caller or system operator. If you



can't think of a good use for this command, you don't need it yet.

- O — Outside. Ordinarily available only to callers with high access levels, the `O)utside` command quits Fido and delivers an errorlevel to DOS. Generally, this command exits to a batch file that lets a caller use programs that live somewhere on the system operator's disk.
- E — Exit to DOS. Ordinarily available only to callers with the same access level as the system operator, the `E)xit-DOS` command quits Fido and delivers an errorlevel to DOS. More often than not, it exits to a batch file that lets the system operator run the system remotely.
- G — Goodbye. Use this to log off the system. (Actually, you can just shut off your modem if you want to.) Often this will give you a chance to leave a private message to the system operator.
- ? — Help. Gets you on-line help, telling what each command does. It's like the list you've been reading, except briefer (usually). Anytime you can't figure out a command in Fido, just type a questionmark and press Enter. Later versions of Fido (version 12 and higher) also have something called contextual help. See "Help Levels" below for more details.

## Changing Caller Settings

To change a setting:

1. Type **C** from the Main Section and press Enter.
2. Type the proper letter for the setting you want to change; then press Enter.
3. Type in the new info.  
If you press Enter without typing anything, the old information remains intact.
4. Repeat steps 2 and 3 for all necessary items.
5. Press Enter without typing anything else when you've finished making changes.

Here's what you can change:

- N — Caller name. Leave a space between your first and last name — that's how Fido tells the difference when you sign on.
- A — City and state. This is the place you usually call from. If you're paranoid, lie. (Computers are stupid and will believe anything.)
- P — Password. Change this every once in a while so that the bad guys can't use your time, or post libelous messages under your name. The system is pretty secure, but realize that the system operator can easily find out your password. Try not to worry about it. (Heh-heh.)
- H — Help level. Fido has three help levels; everybody starts off automatically set to Novice level. For the details, see "Help Levels."
- W — Width of your screen in characters (the `Number of Columns` item).
- L — Length of your screen in lines (the `Number of Lines` item).

M — Turns the [More?] prompt on and off. When this is set to ON, Fido pauses when your screen is full (as determined by the settings under Width and Length, next in this list) and puts the message [More?] at the bottom of the screen: if you type "N" and press Enter, Fido aborts the rest of the message or bulletin and returns you to Command level; if you type any other key and press Enter (or just press Enter without typing anything), Fido gives you another screenfull of text. When More is set to OFF, the text of a message or bulletin just keeps coming, scrolling information off the top of the screen as necessary. When the word More appears without a questionmark (as in [More]), you can't abort the rest of the message.

**Killing a listing:** Usually you can abort any bulletin or message as it's being displayed by pressing Control-C or Control-K. If using Control-C or Control-K doesn't work, the system operator has disabled this option for the particular message or bulletin you're currently reading. The abort keys might work for other messages; continue to question authority.

T — Tab Expansion. If this is set to ON, Fido will send a Control-I to your modem each time it comes across a TAB character in a message or bulletin; if this is set to OFF, Fido sends what it thinks is an appropriate number of spaces for each CONTROL-I it encounters. Tab defaults to ON. If your screen looks OK, ignore this setting; if tables don't line up properly or if you get spurious characters at the far right edge of the screen, change this setting to whatever it's not.

F — Filler-Nulls. Some printing terminals need to be sent null characters for timing purposes. Most terminals don't need any; see the manual that came with your terminal if you experience loss of characters at the beginning of lines.

## Help Levels

Fido has three help levels — Novice, Regular, or Expert. To set the help level, choose the C command from the Main Menu; then type a number for the level of help you want:

1. You're an expert. You have everything memorized. You just want the system to tell you that it's time to give it a command:

Main Command (?=help):

2. You're an experienced caller with a crummy memory. You don't want a lot of help — just a list of all the letters you can type at this command prompt:

MAIN Section:

Commands: M F B S C P L A V G (?=help):

3. You're a novice. You want full help — a one or two-word explanation of each command, plus the list of all the letters you can type at this command prompt:

MAIN Section:

M)sg-Section F)ile-Section B)ulletins

S)tatistics C)hange P)age-Operator

L)ist-Callers A)ns-Questionnaire V)ersion

G)oodbye



Command (?=help):

You can enter a questionmark for more detailed help on a section's commands, no matter what the help level is set at. You can get even more detailed help for a particular command: Type the command's letter immediately followed by a questionmark (for example, `L?`). If you get a message telling you that the help file is missing, leave a message for the system operator saying so; it's probably an oversight.

## Message Section

The message section holds all the messages that callers leave, either directly by typing them into this Fido or indirectly by typing them into the FidoMail section of some other Fido, and sending them to this Fido via FidoNet. Fido can have up to 200 message areas, with an unlimited number of messages in each area. All the commands and prompts in all message areas are exactly the same, except for some special prompts in the FidoNet FidoMail message area. (See "Sending and Receiving FidoMail", later in this chapter, for the details.)

You get to the Message section by entering the M command in the Main Section. Assuming the Help level is set to 3, something close to this appears:

```
Message Area 2: Shred of Dignity Skaters Union
Total messages:    40
Highest message:   53
Highest read:      22

N)ext P)revious E)nter K)ill T)o-You
A)rea-Change R)epl Y)es C)ontinuous L)ist
S)earch U)pdate-Msg O)verride-Path
G)oodbye M)ain-Menu
Command or Message Number 1-53 (?=help) [22 N]:
```

You go to the message area you were in the last time you called this Fido. If you've never given the M command before, you automatically go to the area that the system operator has determined is the place where new message readers go. The title you see above (Shred of Dignity Skaters Union) is the name of this section. Fido is preset to let you read mail as soon as you enter a message section. The N within square brackets at the end of the command prompt [22 N] stands for N)ext, telling you that Fido is ready to display the next message you haven't yet read; all you need do is press Enter.

Fido numbers messages as they're entered. In this example, Message area #2 has 40 messages in it, with the highest message numbered 53. Some messages have been deleted for one reason or another (at least 13 — the difference between 53 and 40), either by the people who entered them, the people they were left for, or by the system operator.

Fido keeps track of the messages you've already read so that you don't waste your time rereading old messages. In the example, the caller has read up to message number 22. If the messages have been renumbered by the system operator to get rid of holes previously occupied by now-deleted messages, the numbers change accordingly. If you haven't been on before, Fido gives you the lowest number it can for the last message you read — the number 1.

Here's what each command does, in brief. The ones that require further explanation have their own detailed sections later on.

N — Next message. This sets the direction of the next and subsequent message numbers that Fido displays for you.. "Next" is defined as 1 + the number that appears after the bracket ([]) just before the end of the prompt. In the example above, the number is 22; so N gets you message #23. If message #23 doesn't exist (because it was deleted) or if #23 is a private message, Fido looks for the next existing non-private message. If no more unread, non-deleted, non-private messages are left, Fido says *Highest message* (which means that you've read all the messages possible to read in this area) and prompts you for another command. To start at the very beginning of the message list in this area and read forward towards the end, type < and press Enter.



P — Previous message. This changes the direction of the next and subsequent message numbers that Fido displays for you. "Previous" is defined as 1 - the number that appears after the bracket ([) just before the end of the prompt. In the example above, the number is 22; so P gets you message #21. If message #21 doesn't exist (because it was deleted) or if #21 is a private message, Fido looks for the next existing non-private message with a lower message number. If no more unread, non-deleted, non-private messages are left, Fido says `Lowest message` (which means that you've read all the messages possible to read in this area) and prompts you for another command. To change the direction, enter the N command. To start at the very end of the message list in this area and read back towards the start, type > and press Enter.

E — Enter a new message. Fido prompts you for appropriate information. Here's an example:

```
This will be message #54
From: Studley Whackburn
To: John Scribblemonger
Subject: Rights of Man
Private? [y,N]: n
Maximum message length, lines: 37
Enter your message, blank line to end
Words will wrap automatically
```

1:

(1: is the number of the line you're about to type; you'll find this line-numbering useful if you need to edit the message after you've written it.) When you enter a message, Fido automatically puts in the message number — always 1 higher than the highest message number in the current area — and fills in the `From:` field with your name.

You fill in the `To:` and `Subject:` fields. (If you press Enter without typing something into either of these fields, Fido assumes you want to cancel the message.) The `To:` field holds up to 35 characters, and the `Subject:` field can hold 71 characters. If you want to make your message unreadable by all callers other than you, the addressee, and the system operator, answer Y to the `Private?` prompt.

**About privacy:** System operators can read private messages. Keep this in mind when you leave messages for people. If the information you want to leave is so private and so personal that you wouldn't want anybody to read it, don't leave it on any bulletin board. This applies doubly to FidoMail, because sometimes FidoMail goes through several boards before it arrives at its destination — and each system operator along the way can read the mail. System operators as a group probably have as much (and as little) integrity as any general cross-section of the population. But even some postal employees find it irresistible to read the occasional postcard.

The system operator decides how many lines long your message can be. You type in your message until you've finished. You can ignore the ends of lines — Fido takes care of carriage returns and line feeds for you (although you can force carriage returns, like at the ends of paragraphs, by pressing Enter). You press Enter twice to let Fido know you've finished. Fido then brings up a set of special editing commands:

```
1: Forget it, John; it's not worth it. There's no way that
2: your mother will let you keep a python in the bathtub.
3:
```

```
S)ave C)ontinue-adding A)bort L)ist E)dit-Line
D)el-Line I)ns-Line H)header (?=help):
```

The commands do what you think they do (Del is short for Delete and Ins is short for Insert). E)dit-Line, D)el-Line, and I)ns-Line all refer to the body of the message; use H)header to edit the information in the To:, Subject:, and Private? fields. (To keep the old information, just press Enter at the appropriate prompt). If you need help with a command, type its first letter and a questionmark, and then press Enter.

**Spelling counts in Fido:** A private message left for *Vladmir Jons* will never be read by *Vladmir Jones*, or by anybody else for that matter (except the system operator). Make sure the spelling in the To: field is correct before you save the message.

Later, when your message has been read by the person you've sent it to, Fido adds the tag (RECV'D) to the message's header line. The number in square brackets before the tag tells you how many times this particular message has been read by anyone coming across it (in this case, 5):

```
#39 26 Jul 87 17:06:31 [5] (RECV'D)
From: Studley Whackburn
To: John Scribblemonger
Subj: Rights Of Man
```

- K — Kills the message you just read, assuming you wrote it or it was addressed specifically to you. You can't kill a message that isn't yours — only the system operator can do that.
- T — Looks for the next message addressed to you — that is, messages that have your name in the To: field. What "next message" is depends on the number and letter that appear between brackets ([ ]) just before the end of the prompt; it will be a message with a higher number than the one in brackets if the letter is N, lower if the letter is P (See N and P, earlier in this section.). To go from the very start of the message list through to the end, enter < T; to go from the end of the message list and work back to the start, enter > T. (Note the space before the T in both cases.)

Fido displays these messages for you one at a time, giving you the chance to reply to a message and then delete it as soon as you've read it. (Some FidOs run on floppy disk, where space is at a premium; so it's always a good idea to throw away a message you no longer need.) Here's what a message retrieved using the T command looks like. In this example, Help level is at 2:

```
Commands: N P E K T A R C L S U G M
or Message Number 1-112 (?=help) [1 N]: t
#81 26 Jul 87 13:12:45 [1]
From: Studley Whackburn
To: John Scribblemonger
Subj: Complaints
```

We don't accept complaints. Sorry.

It's a good idea to reply to a message before you delete it. When you reply to a message you've just read, Fido fills out the TO field for you.

**The bonus RK command:** After you've read a message addressed to you that you want to reply to, use the command RK. The RK command lets you reply to a message, and then automatically deletes the original. That way you don't clutter up the disk with old messages.

- A — Area change. This command shows you all the message areas:



```

----- Message Areas -----
1) Ch@os
2) Shred of Dignity Skaters Union
* 4) FidoNet Messages
5) 'Zine reviews & ads
Message Area [2]:

```

(The area marked with an asterisk (\*) is always the FidoNet FidoMail section.) The number between square brackets at the end of the prompt is the area you were just in, and the area to which you'll automatically return if you press Enter without typing another number.

**About missing numbers:** The numbers of the areas that appear in the areas list might not be sequential. Those areas do exist; you just don't have access to them. Each individual system operator determines who can see what areas and who can use which commands. If you need access to some area that you know exists but that doesn't appear on your screen, leave a note for the system operator asking for access.

**R** — Reply to the message you've just read. You can reply to any message, whether it's addressed to you or not. When you reply to a message, Fido automatically fills out the **From:**, **To:**, and **Subject:** fields for you. (The **To:** field becomes **REPLY TO:**) You need to fill out the **Private?** field yourself. For details on how to enter a message, see the **E** command, earlier in this section.

**Use R rather than E:** Fido keeps track of all replies to a message such that a running conversation can occur. For example, Fred enters a message that Gerry replies to. Alice reads Gerry's reply and replies to it, adding her own comments. Phil sees Alice's reply and adds comments of his own by replying to her message, and so it goes. Later, anyone can trace the conversation using the special **+** command (described later), and read only the series of messages (called the *thread* in BBS terminology) concerned with this topic. To take part in the threaded conversation, always use the **R** command. You can use **E** to write a new message to anybody in the thread, of course; but people who use **+** won't see your comments.

**C** — Continuously show messages. All the messages from the one you last read through the end of the message list scrolls onto your screen without stopping. This happens even if you have "More?" turned on in the **Change** part of the **Main** section. To go from the very start of the message list through to the end, enter **< C**; to go from the end of the message list and work back to the start, enter **> C**. (Note the space before the **C** in both cases.) To stop the list temporarily, press **CONTROL-S**. (Press it again to continue.) To abort the command, press **CONTROL-C** or **CONTROL-K**.

**L** — List the headers (**To:**, **From:**, **Subject:** fields) of all readable messages in the area, starting at the current one and going through to either the start or to the end of the message list, depending on the current direction (**N** or **P**). A "readable message" is one that isn't public, or if private is one that's addressed to or from you. To go from the very start of the message list through to the end, enter **< L**; to go from the end of the message list and work back to the start, enter **> L**. (Note the space before the **L** in both cases.) This command is useful when you want to know who's leaving messages, who's getting messages, or what the subjects of people's messages are. (Of course, this assumes that people — which includes you — leave really descriptive subject lines.)

**S** — Search the header. This command looks in the headers of the current area's messages for

a specified word or phrase that Fido prompts you for. The search starts at the current message and goes through either to the start or to the end of the message list, depending on the current direction (N or P). When it finds a match, Fido displays the header and the rest of the message. To search for the next match, type S and press [CR] twice; Fido remembers the last search string entered.

To go from the very start of the message list through to the end, enter < s; to go from the end of the message list and work back to the start, enter > s. (Note the space before the S in both cases.) To stop the list temporarily, press CONTROL-S. (Press it again to continue.) To abort the command, press CONTROL-C or CONTROL-K.



- U — Update the message you just read. The message must be one that you entered. Use this command to edit a message you've written to someone and already saved. You edit an already-saved message in exactly the same way that you edit a message you're in the process of creating for the first time: Fido displays the message and brings up the editing commands:

```
#86 27 Jul 87 12:52:16 [3] (EDITED)
```

```
From: Studley Whackburn
```

```
To: John Scribblemonger
```

```
Subj: The World Ends Thursday
```

```
1:I have bad news, old chum. The world is coming to an end tomorrow at
```

```
2:exactly 4:15 PM. This is especially annoying, since we had dinner
```

```
3:planned for seven. Sorry.
```

```
4:
```

```
Loaded 3 lines
```

```
You may now display and edit message #86
```

```
S)ave C)ontinue-adding A)bort L)ist E)dit-Line
```

```
D)el-Line I)ns-Line H)header (?=help):
```

Fido adds the comment (EDITED) to the end of the message number line to remind you that you're changing a previously posted message. Later, after you've edited and saved the message, Fido adds an extra line at the start of the body of the message, saying when the editing occurred. This "editing trail" is an integral part of the message; Fido adds it as you save the message, and displays it every time someone reads the message:

```
#86 27 Jul 87 12:52:16 [3] (EDITED)
```

```
From: Studley Whackburn
```

```
To: John Scribblemonger
```

```
Subj: The World Ends Tomorrow
```

```
[NOTE: MESSAGE MODIFIED/EDITED ON 27 Jul 87 12:57:29]
```

```
I have bad news, old chum. The world is coming to an end tomorrow at  
exactly 4:15 PM. This is especially annoying, as we had dinner  
planned for seven. Sorry.
```

- O — Override the normal pathname. Ordinarily available only to callers with a privilege level equal to that of the system operator, the O)verride-Path command lets you specify a different pathname for the display of files summoned by any command in the File section (except the A)rea-Change command). The original pathname is reset as soon as you leave the current area (for example, by using M)ain-Menu, A)rea-Change, or G)oodbye). Most often, this command is used by the system operator to perform Fido housekeeping tasks from some location other than the terminal connected to the computer that's actually running Fido.

- G — Goodbye. Use this to log off the system. (Actually, you can just shut off your modem if you want to.) Often this will give you a chance to leave a private message to the system operator.

- M — Main menu. Returns you to the Main section.

## Special Message Commands

Fido has a few special character commands to help you get around in an area's message section better. A few of them have already been mentioned, but you'll see them again here:

- . (the period) Display (or redisplay) the current message. The current message is the one whose number shows within square brackets (along with the direction indicator N or P) just before the end of the prompt line.
- < Reset the current message number to the lowest in this area and set the direction to N (Next). When you press Enter, the lowest numbered readable message in the area will appear on the screen.
- > Reset the current message number to the highest in this area and set the direction to P (Previous). When you press Enter, the highest numbered readable message in the area will appear on the screen.
- + Read the reply to the last displayed message, if a reply exists. This lets you follow a discussion composed of related messages (a reply to a reply to a reply...) without having intervening messages on other subjects interfere.
- Read the message that this one is replying to. Keep entering the minus sign to get back to the original message that started the discussion. (Fido will let you know when you've reached the original message.) Then you can follow the discussion through to the end by using +.



## **Sending and Receiving FidoMail**

At this writing (August, 1987), Fidos that exchange electronic mail (FidoMail) are operating in 17 countries and in all 50 of the United States. Fidos that exchange FidoMail can connect directly with each other, or they can connect indirectly via "host" Fidos. The public FidoMail network that uses Fido's system of exchanging mail, the International FidoNet Association, currently has over 100 hosts. (A host accepts mail for all Fidos in its immediate geographic vicinity in order to keep telephone costs down.) FidoMail usually gets exchanged in the dark of night, when telephone costs are cheapest.

**How you might use FidoMail:** Assume that you live in San Francisco, and you have a friend or business associate who lives in Boston. S/he has a modem and a computer, and you have a modem and a computer. While costs are determined by each individual system operator, you can write a message of a few hundred words and send it overnight from the Fido that you use in San Francisco to the one that s/he uses in Boston for about 25 cents.

Not all Fidos use FidoNet, the part of Fido that makes exchanging electronic mail possible. To see if the Fido that you regularly call uses FidoNet, use the A)rea command in the Message section: Look for a message area name that has an asterisk (\*) next to it. If no name has an asterisk, then either that Fido isn't using FidoMail or you need to have already set up a FidoNet account before you can even get into the FidoNet area. (See the next section of this manual.) Leave a message for the system operator asking if s/he runs FidoNet and, if so, how you can set up an account. If s/he doesn't run FidoNet, leave a nasty message and call another Fido.

## **Setting Up a FidoNet Account**

Sending an individual FidoNet message is very cheap. (In fact, sending a message to a Fido within your local dialing area is probably free.) But a FidoNet system operator might find that s/he has dozens of long-distance FidoMail messages going out every night. The phone bill for the system operator can add up fast. So most system operators insist that you have an account set up with them before you start sending FidoMail. You'll usually find a bulletin telling you how to set up a FidoNet account. If you can't find the information you need, leave a message for the system operator asking for details.

The Credit line that you see when you enter the Change command in the Main section shows how much you have to spend on FidoMail. The system operator credits your account when you send him whatever s/he requires (or when you fulfill your part of whatever arrangement you have). Your account gets charged for each FidoMail message you make, and the account is reconciled after mail gets sent. You always know how much it costs to send a message before you send it; a list of prices is always available. (You'll find details on how to find prices in the next section.)

## **Using the FidoNet Message System**

(This section assumes you're in the FidoNet message area as you read along.) You use the same process to leave a message in the FidoNet area as you do in any other Message area, except that you also have to add a FidoNet address. Here's what you see after you tell FidoNet you want to enter a message:

This will be message #46

From: Studley Whackburn on Fido 1:125/101  
To:

The number at the end of the `From:` field is the address of the Fido you're calling from. A full address has three sections to it, and it's written in the form *zone:net/Fido*. (The colon and the slash are important.) When you send FidoMail to somebody, you need to include a FidoNet numeric address. To continue the example:

```
This will be message #46
From: Studley Whackburn on Fido 1:125/101
To: Tim Pozar
Current Net is NET      125 SF Bay Net, San Francisco CA, $0.26
Z)one N)et F)ido (or numeric FidoNet address) (?=Help): 1:101/318
```

This piece of FidoMail is going to the Fido board whose complete address is Zone 1, Net 101, Fido 318.

The first number in the address is the *zone* number; it represents some vast geographic area, usually a country. Zone is a feature that's just starting to be used; you can usually ignore it.

The second number (101) represents the *net*, the affiliation of FidOs in a particular geographic area — in the example, the state of Massachusetts (and one or two stragglers from Rhode Island). Note that this number is always followed by a slash.

The third number (318) is the *Fido* number — the number of a specific Fido board. Only one Fido in a net has this number. Somebody in another net might have that same node number, just as somebody in a different telephone area code might have same seven-digit telephone number.

So you can write this complete FidoNet address as either 1:101/318 or simply 101/318.

The problem is how to find the address of the Fido that somebody uses. Unfortunately, there's no international FidoNet caller's directory. Ideally, you know the complete numeric address of the Fido that your friend uses. Short of that, FidoNet can help you find the Fido's address if you know the Fido's net number, its name, or its city. Assume for this exercise that your friend has an account on the PC WEEK board in Boston, Massachusetts.

### The N Command

The N command shows you a list of all the nets:

```
NET      1 Int'l Tech Coord, So Roxana IL, $0.22
NET      10 Calif Nevada, Lompoc CA, $0.26
NET      102 SoCalNet, Los Angeles CA, $0.26
...
...

NET      16 New England, CT ME MA NH RI VT, $0.20
NET      101 MassNet, Gardner MA, $0.20
...
...

NET      713 QLD NET, BRISBANE QLD AUSTRALIA, $2.38
NET      59 OCEANIA INDEPENDENT, SOUTH PACIFIC, $2.38
```



(The ellipses indicate items present in the listing, but left out of this example.) The number on a particular line is the number of the net described. The next item on the line is the name of the host of that net (or a creative contraction of the host's name). The third item is the location of that host. The final number is the amount of money it costs to send a message to that host from the Fido that you're using at the moment. (In most cases, that's also how much it costs to send a message to every Fido served by that host.)

This list shows just a few of the nets; there are over 100 of them. The idea is to browse through the net list until you find one that's likely to encompass the board you're looking for. When you find a likely candidate, you can stop the list by entering CONTROL-C or CONTROL-K.

Assume that the best candidate for a net likely to hold Boston's PC WEEK board is net 101, the MassNet board located in Gardner, Massachusetts. When Fido prompts you, you type

```
101/ [CR]
```

This rather arcane message tells Fido you want to send a message to some board affiliated with net 101. Here's what the prompt looks like after you've searched the Net list and entered the net designation:

```
Current Net is NET      125 SF Bay Net, San Francisco CA, $0.26
Z)one N)et F)ido (or numeric FidoNet address) (?=Help): 101/
Current Net is NET      101 MassNet, Gardner MA, $0.20
Z)one N)et F)ido (or numeric FidoNet address) (?=Help):
```

The current net number has changed; it's now 101. The current net number determines what you'll see when you enter the F command.

### The F Command

This command lists all the FidOs in the current net:

```
Current Net is NET      101 MassNet, Gardner MA, $0.20
N)et F)ido (or numeric FidoNet address) (?=Help): F
NET      101 MassNet, Gardner MA, $0.20
          14 WayStar, Marlboro MA, $0.20
          27 Daves Fido, Gardner MA, $0.20
...
...
          318 PC Week, Boston MA, $0.20
          366 Westboro Fido, Westboro MA, $0.20
...
...
```

Again, there are too many boards in MassNet to list them all, but you don't have to — PC WEEK is Board number 318 in net 101, or 101/318.

Yes, it is indeed crude, but it's effective. A better way of locating a board's address is coming someday. Just hang in there for now.

Try entering some fake FidoMail now to somebody in your own net, just for the experience.

### Sending a File

If you have a high enough access level, Fido will ask if you want to attach a file to your message. You can send any file along with your message, as long as the file exists somewhere

on the system operator's disk. Fido prompts you for a full pathname:

```
Attach File(s)? [y,N]: y  
Full PathName(s):
```

As you can see, you have to have very high access to be able to attach files: Most people don't know what pathnames to use to get around a system operator's disk. (The descriptive names in the File Area list won't do as full pathnames.)

**Why everybody can't attach files:** Most system operators don't let the majority of callers attach files to their messages because of the expense and because of the access to the disk drive(s) this command gives people. As to the expense: It's one thing to send a message of a couple hundred words across the country at 1200 baud; that costs only a few cents. But sending a 50,000-byte file across the country gets real expensive. The price you see in the nodelist only covers a message; it can't take into account the prices for sending files. (At present, FidoNet's accounting system isn't very sophisticated.) Additionally, the Attach Files feature gives a caller full access to the entire disk system. This means, theoretically, that a caller could use MS-DOS wildcard commands to attach the file \*.\* to a message — and send everything on the system operator's disk to Bangkok!



## Files Section

The Files section holds files you can copy for your own use. Callers donate files to the board using the Upload command; the system operator then checks the files to make sure that they're either in the public domain or that the file's copyright owner has given permission that they be distributed without charge. (See the section below called "Honor System Distribution.") Fido can have up to 200 file areas in the Files section, with an unlimited number of files in each area. All the commands and prompts in all file areas are exactly the same.

Here's what you get when you choose **Files** from the Main section:

```
File Area 1: General Rock & Roll
F)iles D)ownload U)pload A)rea-Change T)ype
S)tatistics L)ocate R)aw-Display
K)ill-File O)verride-Path G)oodbye
M)ain-Menu
Command (?=help):
```

(Actually, you're likely not to see several of these commands: `R)aw-Display`, `K)ill-File` and `O)verride-Path` are ordinarily reserved for callers with the same access level as the system operator. They're shown here for completeness.)

The first line tells you the number of the current file area, and gives a brief description of the kinds of files this area holds. The other lines list the commands.

Here's a description of what each command means. Commands preceded by a tilde (~) are ordinarily available only to callers with very high access levels :

**F** — Files list. Shows you the names of the files in this section, the size of each file in bytes, and a brief description of what the file does:

```
CUTEFONE.EXE      7680 Turns your phone # into possible words
AREACODE.EXE     19328 Tells you area codes for cities, and vice versa
QUOTES.BBS       17557 the quotes you get when you log on
CH@OS.AD         1463 Text bomb to drop on other BBSs
```

You can use the MS-DOS wildcard characters \* and ? to list specific files:

Command (?=help): F \*.EXE

```
CUTEFONE.EXE      7680 Turns your phone # into possible words
AREACODE.EXE     19328 Tells you area codes for cities, and vice versa
```

See your MS-DOS manual for the details on wildcard characters.

D — Download one or more files. Use this command to transfer a copy of any file in the Files section to your computer. The system you're using must have some kind of a storage device to hold the file you're getting (which lets you out if you're using just a terminal with no disk drives), and the telecommunications program you're using to call Fido must have a file transfer protocol that Fido can understand. Here's what you see:

```
Command (?=help): D
A)scii, K)ermit, X)modem, XC)modem-CRC, T)elink
Transfer Type: A K X XC T (?=help):
```

Fido tells you the size of the file in blocks (1 block = 128 bytes except for Kermit  
Fido Version 12 77 Copyright 1987 Tom Jennings

where it equals something strange), and how long it will take to download the file using the protocol you've selected. If you don't know what you're doing, use Xmodem.

**Honor System Distribution:** There's a difference between public domain software, "Freeware" (software, documentation, and/or data that the producer is allowing people to use — usually with certain restrictions — without charge), and "Shareware". "Shareware" is software, documentation, and/or data (often copyrighted) distributed on the honor system: You get it free, but you're honor-bound to send payment or a donation to the producer if you continue to use it after a trial period. More often than not, the income from Shareware is the only income the producer gets. So if you don't send the money, you're literally stealing somebody's paycheck.

You can use the MS-DOS wildcard characters \* and ? to download several files in a row if you use the Telink or Kermit protocols. See your MS-DOS manual for the details on wildcard characters; see the manual that came with your telecommunications software for information on Telink and/or Kermit.

U — Upload a file. Use this command to send files for general posting in an appropriate Fido file area. This works essentially like Download, except in reverse. (Fido can't tell how long it will take to upload the file or how many blocks it has, of course, as it does when you download a file.) The system operator will ordinarily check the file you've sent to make sure that it's either in the public domain or that the file's copyright owner has given permission that the file be distributed without charge.

You can use the MS-DOS wildcard characters \* and ? to upload several files in a row if you use the Telink or Kermit protocols. See your MS-DOS manual for the details on wildcard characters; see the manual that came with your telecommunications software for information on Telink and/or Kermit.

**Got enough room?** Use the Statistics command in the Files section to check the available free space on the Fido you're using before you use the Upload command. Fido has no way to warn you in advance that your upload might get chopped off in the middle — which happens when Fido runs out of upload disk space.

A — Area change. This commands shows you all the file areas and lets you change to another one:

```
----- File Areas -----
1) General Rock & Roll
* 2) FidoNet Files
3) Night Club Action
4) IBM-PC Software
5) Macintosh Software
File Area [3]:
```

(The area marked with an asterisk (\*) is always the FidoNet Files section.) The number between square brackets just before the prompt is the area you were just in, and the area to which you'll automatically return if you press Enter without typing another number. Type the number of the file area you want to move to and press Enter.



**About missing numbers:** The numbers of the areas that appear in the areas list might not be sequential. Those areas do exist; you just don't have access to them. Each individual system operator determines who can see what areas and who can use which commands. If you need access to some area that you know exists but that doesn't appear on your screen, leave a note for the system operator asking for access.

**T** — Type a text file on screen. This command scrolls the contents of a text file onto your screen. (Files with the suffix .DOC, .ASC, and .TXT are almost always text files.) It's essentially the same as downloading a file using the ASCII protocol.

**S** — Statistics. This shows the same information as does the statistics command in the Main section, except that files information replaces the FidoNet credit information. Specifically, you'll see information on the number of bytes you've uploaded or downloaded, and the space available for uploading files in this area:

```

28 Jul 87  21:13:22
                Total   Limit   Remaining
Total Calls:      264
Your Calls:       41
Connect Time:     1           0           0
24 Hr Time Limit:           0           0
Disk Space:              20M        17M
Downloaded:         0
24 Hr Dnld Limit:           0K        0K
Uploaded:          0K

```

**L** - Locate a file in any file area. Use this command to find out if Fido has a file you're looking for, and if so what area it's in:

```

Command (?=help): L
File(s) to search for: TWIX.DOC
  1) General Rock & Roll
*  2) FidoNet Files
  3) Tools
TWIX.DOC      774 accurate but ugly
  4) IBM-PC Software
  5) Macintosh Software

```

Found 1 matching files

You can use the MS-DOS wildcard characters \* and ? to locate specific files. See your MS-DOS manual for wildcard character details.

**~R** — Ordinarily available only to callers with high access levels, the **R)aw-Display** command lists all files and subdirectory names in the current directory. (The Files command in reality just reads a file called FILES.BBS, an ordinarily hidden file that holds a list of essential data about all available files, but that doesn't necessarily show the complete contents of any subdirectory.) Most often, this command is used by the system operator to perform Fido housekeeping tasks from some location other than the terminal connected to the computer that's actually running Fido.

You can use the MS-DOS wildcard characters \* and ? to locate specific files. See your MS-DOS manual for wildcard character details.

**~K** — Ordinarily available only to callers with a privilege level equal to that of the system operator, the **K)ill-File** command deletes a file and removes its entry from FILES.BBS. (The contents of FILES.BBS is what gets displayed when you give the File

command.) Most often, this command is used by the system operator to perform Fido housekeeping tasks from some location other than the terminal connected to the computer that's actually running Fido.

You can use the MS-DOS wildcard characters \* and ? to delete files. See your MS-DOS manual for wildcard character details.

- ~O — Override the normal pathname. Ordinarily available only to callers with a privilege level equal to that of the system operator, the O)verride-Path command lets you specify a different pathname for the display of files summoned by any command in the File section (except the A)rea-Change command). The original pathname is reset as soon as you leave the current area (for example, by using M)ain-Menu, A)rea-Change, or G)oodbye). Most often, this command is used by the system operator to perform Fido housekeeping tasks from some location other than the terminal connected to the computer that's actually running Fido.
- G — Goodbye. Use this to log off the system. (Actually, you can just shut off your modem if you want to.) Often this will give you a chance to leave a private message to the system operator.
- M — Main menu. Returns you to the Main section.



## Fido's Command Buffer: An Undocumented Feature

Experienced callers can save a lot of time by skirting many of Fido's prompts. You do this by putting several commands on the same line, separated by spaces. For example, assume that you start from the Main Section, and that you want to list the files in area 7 — an area different from the one you looked at the last time you used the File section. Here's how you'd do it:

```
F A 7 F [CR]
```

You'll go directly from the Main Menu to a list of the files in Area 7. This is possible because Fido's commands are all (or nearly all) one-character commands, a bunch of which Fido can remember in its command buffer.

Here's a further shortcut: Use a vertical bar to represent a carriage return. For example, at the signon prompt, Fido expects to see up to three names (First, Middle, Last). Use the vertical bar to indicate you've finished with your name, then go on to confirm that the name is correct and add your password:

```
John Scribblemonger | Y Scribe [CR]
```

You'll end up going directly from the signon prompt to the post-signon messages without going through the prompts in between.

The More? prompt clears Fido's command buffer. This means that if a system message is so long that it puts the More? prompt at the bottom of the screen, Fido waits for you to press the Enter key before going on — and then forgets whatever other commands you've entered. For example:

```
John Scribblemonger | Y Scribe F A 7 D X Mortcalc.EXE | G N [CR]
```

This line would log John Scribblemonger onto the system (using his password Scribe), go to area 7 of the Files section, use the Xmodem protocol to download a program called Mortcalc.EXE, and log off the system without leaving a message for the system operator. But if the system operator has a long post-signon message that invokes [More?], all the commands after the password get trashed. The solution is to use the C)hange command to turn off the [More?] prompt — and pray for a clean phone line.

**Just to be complete:** Other things that flush this type-ahead buffer (as it's called in the trade) are CONTROL-C, CONTROL-K, CONTROL-S (which pauses output), and CONTROL-F (whose only purpose is to flush the type-ahead buffer).

## Appendix A- Delta Guide to Version 12

Written by  
Tom Jennings

This appendix covers upgrading from Version 11 to Version 12, and is in three parts:

- Using the 11-TO-12.EXE program to convert your installation
- A list of the major enhancements and changes to Version 12
- A table listing the files that have changed from Version 11 to Version 12

### Converting from Version 11 to Version 12

The "11-TO-12.EXE" program is for converting a Fido version 11 installation into a nearly complete Version 12 installation. It is only "nearly" complete because you still need to edit a few text files, and you really should look at the new features in the .INI files while you are there. In any case, the whole process should take a matter of minutes.

11-TO-12 does all the dirty work of the conversion process:

- Converts all of the SYSTEMn.BBS files (that contain Message and File Area information and privilege levels) into a text file V11AREAS.INI that you can add to the supplied AREAS.INI. It will go into each Area and read the DIR.BBS file to find that Areas description.
- Converts existing files into new formats and names: the caller file USER.BBS, welcome messages, log files and other system files.
- Converts scheduled events into the text file V11EVENTS.INI so you can incorporate it into the new scheme of things.
- Converts your current batch file command line switches into a text file V11FIDO.INI for you to add to the supplied. FIDO.INI.

All of your Message and File Areas will be usable as-is, with no changes necessary at all; 11-TO-12 will know how to convert the subdirectory names.

11-TO-12 will NEVER modify any existing Version 11 files, so it is safe to run any number of times.

### Running 11-TO-12.EXE

1. **Create a new subdirectory to contain your new Version 12 Installation.** Later on you can delete your Version 11 installation, but wait until you are satisfied with the conversion process first.
2. **Copy all of the programs and files from the distribution diskette into the new Version 12 directory.** Do this FIRST, because 11-TO-12 will modify or replace sample files such as the caller list and welcome messages with new ones properly converted from your working Version 11 installation.



**3. Copy 11-TO-12.EXE to your current Version 11 directory and run it.** 11-TO-12 will ask you for the subdirectory you want to put Version 12 into, and create it if it doesn't already exist. 11-TO-12 prompts you at each step; enter "Y" or "N" followed by [CR].

**4. When 11-TO-12 is complete, log into the new Version 12 directory.**

Once you have completed all the conversion steps, you will be left with the intermediate .INI files: V11FIDO.INI, V11AREAS.INI and V11EVENTS.INI. It is recommended that you edit the .INI files supplied with Version 12 and add in the important parts from the Version 11 .INI files 11-TO-12 makes; the files supplied contain much more information and options than 11-TO-12 generates.

Many of the file names you are used to seeing have changed; the new names and additions are described in "Appendix B: Files and their Functions".

Look through the supplied .INI files and adjust things as you see fit; if you are familiar with Fido, most of the things in these files will be very familiar, and you will probably be able to guess their purpose and operation immediately.

Proceed with the installation process in Chapter 1; you can probably just skim it and look at the tables.

## File Name and Format Changes

These are the files that have changed in Version 12. Some only have different names; some have different formats or functions. Refer to "Appendix B: Files and their Functions" for a complete list of files.

current name	old name	what changed
NEWCLR1.BBS	NEWUSER1.BBS	name only
NEWCLR2.BBS	NEWUSER2.BBS	name only
NEWCLR.QES	QNEWUSER.BBS	name only
NEWCLR.ANS	ANEWUSER.BBS	name only
NOPWD.QES	QNOPWD.BBS	name only
NOPWD.ANS	ANOPWD.BBS	name only
BULLETIN.1	BULLETIN.BBS	first line is now displayed title
PAGE.BBS	YELL.BBS	name only
FIDO.LOG	SYSOP.LOG	name only
MAIN.HLP		internally formatted; see Chapter 1
CHANGE.HLP		internally formatted; see Chapter 1
MESSAGE.HLP	MSG.HLP	internally formatted; see Chapter 1
NODES.HLP		internally formatted; see Chapter 1
EDIT.HLP	ENTERCMD.HLP	internally formatted; see Chapter 1
FILE.HLP		internally formatted; see Chapter 1
XFERTYPE.HLP		internally formatted; see Chapter 1
NODELIST.SYS		internal format change
NODELIST.IDX		internal format change
FIDONET.LOG	MAILER.LOG	name only
CALLER.SYS	USER.BBS	format change

These files are no longer used; they have been replaced by other functions.

obsolete file	replacement file or function
SYSTEM.BBS	functions moved to FIDO.SYS
SYSTEM??.BBS	same as SYSTEM.BBS
DIR.BBS	functions moved to AREAS.INI
SCHED.BBS	functions moved to EVENTS.INI
MAINPRIV.BBS	functions moved to COMMANDS.INI
MSGPRIV.BBS	functions moved to COMMANDS.INI
FILEPRIV.BBS	functions moved to COMMANDS.INI
CHGPRIV.BBS	functions moved to COMMANDS.INI
FIDOMDM.BBS	variable modem-string in FIDO.INI
MAILSYS	functions moved to FIDO.INI
LASTUSR.BBS	functions moved to FIDO.SYS



## Appendix B- Files and their Functions

Written by  
Tom Jennings

This is a comprehensive list of all the files that Fido uses. (It also implies that you shouldn't use these names for anything else, at least in the subdirectory you devote to Fido.) Chapter 1 describes each file in detail; this table is just for reference.

Files marked with a  $\sqrt$  have different names modified when n/l is put on the command line; refer to Chapter 2, "Command Line Switches", for details.

### Installation Files

FIDO.INI	Main initialization file
AREAS.INI	Message and File Area definitions
EVENTS.INI	event definition
COMMANDS.INI	command names
MSGMGR.INI	Message Manager instructions

### Text files displayed during various parts of the signon process

WELCOME1.BBS	first welcome file a caller sees
WELCOME2.BBS	displayed after password complete; right before Main Menu
NEWCLR1.BBS	second file a new caller sees; before "where are you calling from?"
NEWCLR2.BBS	for new callers, right after "choose a password", if no NEWCLR.QES
NEWCLR.QES	optional new-caller questionnaire; overrides NEWCLR2.BBS
NOPWD.BBS	new callers & private system, or forgotten password, and no NOPWD.QES
NOPWD.QES	optional questionnaire; overrides NOPWD.BBS

### Optional bulletin files displayed by the B)ulletins command

BULLETIN.1	first bulletin,
BULLETIN.2	second bulletin,
...	...
BULLETIN.99	up to 99 bulletins, ie. "BULLETIN.99"

### Misc. text files

PAGE.BBS	Displayed when operator not available; P)age command
DNLDL.BBS	Displayed when the caller has exceeded the set k-limit
QUOTES.BBS	optional quotations, displayed right before Main Menu
$\sqrt$ FIDO.LOG	where caller transactions get logged
MAINSECT.QES	A)ns-Questionnaire command
MAINSECT.ANS	answers to above
NEWCLR.ANS	answers to NEWCLR.QES
NOPWD.ANS	answers to NOPWD.QES

## Help Files

MAIN.HLP	Main Menu help file
CHANGE.HLP	C)hange command, in Main Menu
MESSAGE.HLP	Message Menu help
NODES.HLP	E)nter, R)eply or U)pdate, inside "Choose a Fido"
EDIT.HLP	E)nter, R)eply or U)pdate help, inside Message Menu
FILE.HLP	File Menu help
XFERTYPE.HLP	D)ownload and U)pload file transfer protocol help

## FidoNet related files

NODELIST.BBS	the master list of FidoNet nodes in your network
NODELIST.SYS	compiled nodelist database
NODELIST.IDX	database index file
NODELIST.NMP	database node map statistics
NODELIST.NTC	database list of outgoing packets
ROUTE.BBS	default text file of route language statements
ROUTE.A	route language file for FidoNet "A"
...	route files ROUTE.B through ROUTE.W
*.OUT	outgoing packet files
*.FLO	outgoing packet files, lists of attached files
*.IN	incoming packet files
*.FLI	incoming packet files, lists of attached files
√FIDONET.LOG	where FidoNet activity gets logged

## Special format files

FIDO.SYS	main Fido system file
CALLER.SYS	the database of callers to Fido
CALLER.OLD	purged caller records, created by SYSOP.EXE
CALLER.BAK	backup caller database, created by SYSOP.EXE
TIMELOG.BBS	caller statistics; see TIMELOG.EXE
FILES.BBS	one in each File Area subdirectory; holds the names of files
√NODES.BBS	temporary file created by "choose a Fido" in Message Menu
*.MSG	message files contained in each Message Area subdirectory



## Appendix C- Modem Basics

Written by  
Tom Jennings

Fido works only with "Hayes-compatible" modems, that respond to "AT" commands by performing actions and returning result codes back to Fido. During normal operation, Fido treats all modems as basically the same, issuing the same "AT" commands regardless of what kind of modem is installed.

### Modem Initialization

Before doing anything else with the modem, Fido initializes it so that it performs in a predictable manner. If the initialization fails, because you have the wrong `modem-type` set in FIDO.INI; because the modem is connected improperly; or because your modem is incompatible, then Fido and FidoNets behavior will be unpredictable.

You might have an installation where Fido's initialization doesn't work just right; for example, if your modem is connected to a private PBX, you may need to set the wait-for-connect delay much longer than normal. The variable `modem-string` in FIDO.INI provides for this; this special initialization is discussed in detail below.

The modem initialization process is done when:

- Fido is first run
- When a FidoNet event starts
- System operator presses the "I" key

The initialization process is the same for all modems; only the particular commands issued varies from modem to modem. The initializations steps are:

1. **Choose the initial baud rate (see table below)**, which (by no coincidence) is the highest speed the modem can operate at. This speeds up the initialization process.
2. **Issue one Carriage Return [CR]**. This clears the modem's command line buffer, in case any "garbage" was left by preceding programs.
3. **Perform a simple "modem check" function**: modem check consists of issuing a plain "AT" command to get the modems attention; "AT" by itself performs no action other than to cause the modem to return a result code of "OK". Fido tries up to three times to get the modems attention with a plain "AT", after which it displays the error message " \* WARNING: Modem not responding; check modem installation" and continues with the rest of the initialization process anyway.
4. **Issue the basic commands "ATV0Q0E0S0=0M0"**. (See the table immediately below for details.) Fido waits for the modem to respond with a result code of "0" (numeric equivalent of "OK") before proceeding to the next step. If the modem doesn't respond within four seconds, Fido continues anyway. The command break down is:

command	meaning
AT	Attention command
V0	numeric result codes
Q0	send result codes
E0	no command echo
S0=0	do not auto-answer
M0	turn the speaker off

**5. Fido then issues another command string dependent on the modem-type variable in FIDO.INI.** (See the table below) Fido waits for a result code or four seconds to elapse as before.

**6. If there is any special initialization to be done (the variable modem-string in FIDO.INI) it is done now.** Fido outputs the modem-string exactly as it is defined in FIDO.INI, then waits for result codes as in steps 4 and 5, outlined above. If modem-string is undefined, this step is skipped.

**7. The baud rate is set to 300 baud.** (the lowest baud rate in common with all modems). Fido normally converses with the modem at 300 baud. The modem announces events such as "Ring!" and Fido issues the proper commands to answer the phone all at 300 baud.

**8. Fido performs the "modem check" process as defined in Step 3.** Fido will report the same error if the modem still won't respond.

After completing these steps, the modem is assumed to be ready to used, and Fido goes about it's normal business.

#### **Modem Initialization (see Step 3 and Step 5, above)**

modem-type	maximum baud rate	commands issued
1	1200	ATX1
5	1200	ATX1
7	1200	ATX1S18=0
11	2400	ATX3&D2&C1&W
12	2400	ATX3
14	9600	ATX3&W
10	2400	ATX1S17=32



## Going OnHook (Disconnecting)

Fido performs a surprising number of steps to accomplish the seemingly-simple task of hanging up on somebody. Some of the steps are similar ones done in initialization. OnHook is performed when:

- Caller logs out with G)oodbye
- Fido logs out a caller because their time limit ran out
- FidoNet has completed a dial/connect operation

The disconnect process involves the two modem signal lines "CD" (Carrier Detect, usually Pin 6 or Pin 8) and "DTR" (Data Terminal Ready, usually Pin 20):

- The modem sets the CD signal TRUE when it is connected to another modem
- Fido sets DTR ON to enable the modem; OFF to disable the modem

Fido requires that the CD signal be present and working; Fido prefers that the DTR signal be present and working, but it will operate without it. Fido assumes that when the "CD" signal is TRUE, that the modem is connected to a remote modem. CD is the only way that Fido has to tell if the modem is connected or not.

The purpose of the `cd-bit` variable in FIDO.INI is so that Fido can read the "Carrier Detect" signal carried from the modem to the Async Adapter in your computer. Fido logically ANDs the contents of the Modem Status Register with the `cd-bit` value, and if the result is not zero, Fido assumes the "CD" signal is TRUE.

The disconnect process performs the following steps:

**1. Fido waits until the output buffers have drained into the modem.** Fido sends characters destined for the caller (via the modem) to an "output buffer"; on a background-interrupt basis characters are moved from the buffer to the modem. (This is because the modems send and receive at slow, fixed rates such as 30 or 120 characters per second, and software like Fido operates much faster than that. Having output buffers means Fido doesn't need to hang around waiting for the slowpoke modem, but can just stuff characters into the buffer and go about it's business. (All modems, by definition, are too slow, even 9600 baud ones.))

This keeps Fido from disconnecting the caller in mid-sentence. If for some reason the modem is already disconnected, the buffers are immediately emptied; since there is no longer a caller, throwing out the queued up data is only sensible.

**2. Fido issues a plain [CR] character.** This empties any characters that might have accumulated in the modem's command buffer.

**3. Fido tries Disconnect Method #1.** This consists of turning off the Modem-Enable signal "DTR". (Data Terminal Ready, usually on Pin 20) Most modems will immediately disconnect when this signal is turned off. Fido waits up to 1 second for the modem to disconnect with DTR. (DTR will stay OFF for at least 100mS.) The DTR signal is then turned on again. If this step was successful, Fido proceeds to step 6.

**4. If the modem is still online, Fido tries Disconnect Method #2.** "+++" commands are issued, followed by "ATH0" if necessary, with the appropriate "guard time" as required by the modem. This is the standard "disconnect" method for Hayes type modems. Fido will attempt this five times. If successful, Fido proceeds directly to step 6.

**5. If the modem is still connected (if "CD" is still TRUE) then Fido presumes that there is a serious problem** and displays a large warning and continually tries both disconnect methods until the system operator types Control-C to abort Fido. Fido will not return to normal operation from this point; manual intervention is required.

**6. 300 baud is selected, and a "modem check" is performed** as described earlier. This ensures that the modem has completed the disconnect and is ready for further commands.



## Index

- \*.MSG 86
- \*.TLG 86
- .INI files, customizing 6
- .INI files, making changes 32
- .INI files, when changes take effect 32
- .INI files, from version 11 82
- 11-TO-12.EXE program 82
- ?=help command 63, 65
- @ magic file name 41
  
- A)ddress command inside C)hange 62
- A)ns-Questionnaire command 27, 62
- A)rea-Change command 68, 78
- access controls 12, 14, 18, 29, 31, 34, 40, 43, 61
- AREAS.INI 34, 52, 85
- AREAS.INI (message and file area definitions) 13
- ASCII protocol 77
  
- B)ulletins command 61
- batch files 2, 9
- baud rate, modem maximum 10
- bulletin files 85
- BULLETIN.BBS 24, 85
- BULLETIN.nnn files 24, 85
  
- C)hange command 60, 62, 63
- C)ontinuous message read command 69
- caller database 43
- caller default message and file area 15
- caller statistics 47
- caller time limits 9, 22
- CALLER.BAK 45, 86
- CALLER.OLD 45, 86
- CALLER.SYS 43, 45, 86
- callers guide 58
- callers, removing from the database 45
- callers, first time on 33
- callers, disconnecting them 38
- callers, removing from the database 46
- Carrier Detect signal 10, 89
- CD-Bit variable (Carrier Detect) 8, 10
- CHANGE.HLP 25, 86
- changing command names 16
- chat with the caller 37
- choosing a password 33, 60, 63
- COM1 10, 39
- COM2 10, 39
- command line switch /I and file names 85
- command line switches 39
- command names 16, 34
- command names and help levels 65
- command names, initial settings 19
- command-ahead buffer 81
- COMMANDS.INI 85
- COMMANDS.INI (command names, locks and privilege levels) 16

- communications programs 59
- compatibility
  - hardware 2
  - modems 1, 10, 87
  - software 2
- Connect-Tries variable (FidoNet dialer) 8, 10
- console commands 37
- Control-C 24, 64, 75, 81
- Control-F 81
- Control-K 24, 64, 75, 81
- Control-S 81
- costs of FidoNet messages 73
- country-dependent variables 8, 11, 16
- credit and debit 73
- customizing Fido 6
- customizing system notices 24, 85
  
- D)ownload command 40, 77
- Daily-Limit (daily time limit) 8
- Daily-Limit variable (daily time limit) 9
- Data Terminal Ready signal 89
- days of the week 21
- DEC Rainbow 10
- Decimal variable 8, 11
- Default-Priv variable 31
- Default-Priv variable (new caller initial privilege) 8, 12
- Dial-Tries variable (FidoNet dialer) 8, 10
- DIR.BBS 82
- disk space 2
- DNLDL.BBS 85
- DOS access by callers 9
- DOS commands 2
- DOS, terminating to 9
- download controls and limits 9, 14
- download file sizes 40
- download files 58
- download limits 9
- download protocols 77
- downloadable files and descriptions 40
- downloading multiple files 78
  
- E)nter command 67
- E)xitDOS command 9, 63
- E-Errorlevel (ExitDOS command) 9
- E-Errorlevel variable (ExitDOS command) 8
- EDIT.HLP 25, 86
- editing messages 67
- electronic mail 49, 52, 58, 73
- entering FidoNet messages 74
- errorlevel variables in FIDO.INI 8, 9
- errorlevel variables in FIDO.INI 22
- errors, where they are logged 48
- event execution 23
- event windows 21
- events 21, 48, 49, 50
- EVENTS.INI 21, 33, 38, 85
- events, listing upcoming 33



events, FidoNet 53  
events, FidoNet 52  
execution at sunrise 23

F)ido command 75  
F)ile-Section command 61  
F)iles command 40, 77  
F)iles command options 77  
F)iller-Nulls command inside C)hange 62  
FIDO.INI 39, 82, 85, 87  
FIDO.INI (general system parameters) 7  
FIDO.LOG 48  
FIDO.SYS 86  
FidoMail 52  
FidoNet 49, 58, 73  
FidoNet addresses 8, 10, 51, 52, 74  
FidoNet addressing shortcuts 75  
FidoNet dialing controls 10  
FidoNet events- defining 22  
FidoNet, how it works 52  
FidoNet message and file areas 15  
FidoNet messages 66  
FidoNet networking controls 8, 10  
FidoNet topologies see topologies  
FidoNet variables 8, 10  
FidoNet work files and reserved file names 86  
FidoNet- how it works 52  
FidoNet- what it does 52  
FIDONET.LOG 86  
FidoNet, what it does 52  
file area definitions 13  
file areas 18, 34, 58, 77, 78, 82  
file areas, seperate upload and download 41  
file descriptions 40  
file names and command line switch /I 85  
file transfer protocols 77  
FILE.HLP 86  
FileArea keyword in AREAS.INI 13  
files and their functions 85  
Files Section 77  
FILES.BBS 40, 41  
FILES.HLP 25  
files, newly uploaded 41  
files, list of available 40  
file, creation time and date 41  
filler characters 64  
First-Limit variable (first time caller time limit) 9  
  
G)oodbye command 13, 15, 31, 34, 63, 71, 80, 89  
G-Errorlevel (Goodbye command) 9  
G-Errorlevel variable (Goodbye command) 8  
  
H)elp-Level command inside C)hange 62  
hardware supported 1  
help files 25, 34, 64, 86  
help files, their contents 26  
help, in Main Section 34

help levels 64  
help system- how it works 26, 34, 64, 65  
hobbiest networks 51

IBM PC and compatibles 10  
Idle events 22  
IFNA 3, 49, 51, 73  
initializing the modem 32  
installation 4, see Setup  
IO-Port variable (Async Card selection) 8, 10

K)ill command 68  
K)ill-File command 79  
K-Limit (download limit) 8  
K-Limit variable (download limit) 9  
Kbyte limits 9  
KERMIT protocol 77

L)ength command inside C)hange 62  
L)ist messages command 69  
L)ist-Callers command 62  
L)ocate command 79  
local login to Fido 39  
local screen output 37  
locks and keys 14, 16, 18, 34, 43, 44, 61  
locks on message and file areas 13  
log files 48  
logging in the first time 60  
logging into Fido 59  
login from local keyboard 33, 37  
loops see non-terminating loops

M)ain-Menu command 71, 80  
M)ore command inside C)hange 62  
M)sg-Section command 61  
Mail-Errorlevel (Unscheduled Mail) 9  
Mail-Errorlevel variable (Unscheduled Mail) 8  
Main Section command descriptions 61  
MAIN.HLP 25, 86  
MAINSECT.ANS 29, 85  
MAINSECT.QES 29, 85  
MAKELIST.EXE program 57  
memory requirements 1  
message and file areas 18, 82  
message area definitions 13  
message area names 66  
message areas 34, 58, 66  
message length 12  
message section 66  
Message Section command menu 66  
MESSAGE.HLP 25, 86  
messages  
    editing 67  
    editing existing 71  
    entering network messages 73  
    F)ido command 75  
    FidoNet 66



- FidoNet costs 73
- FROM
  - field 67
  - how to read and write 66
  - maximum length 67
  - N)et command 75
  - privacy issues 67
  - purging old 46
  - reading and writing 66
  - special commands for reading 71
  - SUBJECT field 67
  - TO field 67
  - writing 73
- Z)one command 75
- modem variables 8, 10
- modem basics 87
- Modem-String variable (custom modem initialization) 8, 10
- modem-string variable in FIDO.INI 87
- Modem-Type variable 8, 10
- modem-type variable in FIDO.INI 87
- modems
  - AT commands 87
  - Carrier Detect signal 89
  - compatibility 87
  - custom initialization 88
  - Data Terminal Ready control 89
  - dialing 56, 59
  - disconnecting 89
  - Hayes and compatibles 10
  - initialization 35, 37
  - initialization commands 88
  - initialization process 87
  - maximum baud rate 87
  - Multi Tech 10
  - Prentice 10
  - runtime checks 87
  - special installations 8
  - status and controls 89
  - US Robotics 10
  - US Robotics, Courier, HST 10
- modems, special installations 10
- modems supported 10
- Money variable 8, 11
- Msg-Length variable 8, 12
- MsgArea keyword in AREAS.INI 13
- MSGMGR.EXE program 46
- MSGMGR.INI 46, 85
- multitasker programs 37, 39
  
- N)et command 75
- N)ext command 66
- Net and Node addresses 8, 10
- Net variable (FidoNet address) 8, 10
- nets 49, 50, 74
- network addresses 51, 52
- network topologies see topologies
- networks 49

- net, what it means 51
- new caller questionnaires 27
- new callers 33
- New-Caller-Key variable 8, 12
- NEWCLR.ANS 85
- NEWCLR.QES 29, 31, 85
- NEWCLR1.BBS 24, 31, 33, 85
- NEWCLR2.BBS 24, 29, 31, 33, 85
- Node variable (FidoNet address) 8, 10
- nodelist
  - compiling 57
  - format and contents 56
  - sample file 57
  - what it does 56
- nodelist files 86
- NODELIST.BBS 56, 57, 86
- NODELIST.IDX 57, 86
- NODELIST.NMP 57, 86
- NODELIST.NTC 86
- NODELIST.SYS 57, 86
- nodes 49, 50, 74
- NODES.BBS 86
- NODES.HLP 25, 86
- node, what it means 51
- nodes, network host 51
- non-terminating loop see loop
- NOPWD.ANS 85
- NOPWD.BBS 29, 30, 85
- NOPWD.QES 29, 85
- Normal-Limit (caller time limit) 8
- Normal-Limit variable (caller time limit) 9
  
- O)utside command 63
- O)utside command errorlevel 9
- O)verride-Path command 71, 80
- O-Errorlevel (Outside command) 9
- O-Errorlevel variable (Outside command) 8
- OPTional events 22
  
- P)age-Operator command 25, 38, 62, 85
- P)age-Operator command- enabling it 21, 23
- P)age-Operator command, enabling it 21
- P)assword command inside C)hange 62
- P)revious command 66
- PAGE.BBS 25, 85
- Path-Display variable 8, 12
- pathnames, message and file areas 13
- previous Fido versions see Version 11
- private systems 27, 29, 30, 31
- Private variable (Private vs. Public) 8, 12
- privilege level 4 and higher 41, 75
- privilege levels 14, 16, 18, 31, 34, 36, 43, 61
- privilege levels, initial settings 19
- privilege levels on message and file areas 13
- public bulletin board networks 51
- public vs. private 29, 30, 60



- questionnaires
  - MAINSECT.QES 29
  - making your own 27
  - multiple choice 27
  - NEWCLR.QES 29
  - the answer file 28
- quotations 25, 33
- QUOTES.BBS 25, 33, 85
- R)aw-Display command 79
- R)eply command 69
- region, what it means 51
- ROUTE.A 86
- ROUTE.BBS 86
- routing
  - ACCEPT-FROM command 53
  - ALL substitute word 55
  - examples 53
  - EXTERNAL-MAIL command 54
  - FidoNet default 51
  - HOLD command 54
  - HOSTS substitute word 55
  - MYNET substitute word 55
  - NET command 54
  - NO-ROUTE command 54
  - NOT negation 55
  - PICKUP command 54
  - POLL command 54
  - RECEIVE-ONLY command 54
  - route file commands 53
  - route file names 86
  - route files 53
  - ROUTE-TO command 53
  - SEND-ONLY command 54
  - SEND-TO command 53
  - THISNET substitute word 55
  - what it does 52
  - ZONE command 54
- running Fido 36
- S)earch command 69
- S)tatistics command 61, 79
- scheduled events 9, 21, 33, 38, 48, 49, 50, 52
- scheduled events and caller time limits 22
- scheduled events, displaying 37
- semi-private systems 31, 60
- serial interface 10
- SET-FIDO.EXE program 32
- setting up Fido 4
- setup
  - .INI startup files 6
  - AREAS.INI 6, 13
  - basic questions 5
  - command line switches 39
  - COMMANDS.INI 6, 16
  - customization 6
  - EVENTS.INI 6, 21

- FIDO.INI 6
- FIDO.INI startup file 7
- first steps 5
- modems 32
- note to veteran system operators 6
- preset values 6
- previous version considerations 6
- public or private? 5
- setting your privilege level to 7 45
- system parameters 6
- Signon-Limit (log in time limit) 8
- Signon-Limit variable(log in time limit) 9
- simultaneous keyboards 37, 38
- special characters 81
- special format files 86
- subdirectories 2, 13, 32
- subdirectories, creating new ones 13
- sysop see system operator
- sysop-level access 35
- SYSOP.EXE program 30, 43
- system activity log 47, 48
- system notice files 24, 85
- system notices 64
- system operators job 36, 43, 48, 49
- system parameters 6
- SYSTEMnn.BBS 82
  
- T)o-You command 68
- T)ype command 79
- tab characters 64
- TELINK protocol 77
- terminate Fido after current call 37
- test mode 33, 39
- Thousands-Separator variable 8, 11
- time controls and limits 9
- TIMELOG.BBS 47, 86
- TIMELOG.EXE program 47
- topologies
  - amorphous 50
  - thre-tiered 50
  - two-tier 50
  
- U)pdate-Msg command 71
- U)pload command 40, 78
- undocumented feature 81
- unscheduled mail errorlevel 9
- upgrading from version 11 5, 24, 39, 82
- upgrading from version 11- files that changed 84
- upload controls 14, 15
- upload file area 13
- upload files 58
- upload subdirectory 13
- uploaded file descriptions 40
- uploads 40
- USER.BBS 82
  
- V)ersion command 63



V11AREAS.INI- upgrading from version 11 82  
V11EVENTS.INI- upgrading from version 11 82  
V11FIDO.INI- upgrading from version 11 82  
version 11 message and file areas 82  
version 11, obsolete files 84  
version 11 upgrade- files that changed 84  
version 11 upgrade- the process 82  
vertical bar character 81  
Victor 9000 10

Width command inside Change 62  
Waiting for a call or event 32  
WELCOME1.BBS 24, 33, 85  
WELCOME2.BBS 24, 33, 85  
wildcard characters 78, 79  
writing messages 67

Xfer-Display variable 8, 12  
XFERTYPE.HLP 25, 86  
XMODEM protocol 77

Zone command 75  
Zone variable (FidoNet address) 8, 10  
zones 49, 50, 74  
zone, what it means 51

1. The first part of the report is a general  
description of the project and its objectives.  
2. The second part is a detailed description of  
the methodology used in the study.  
3. The third part is a description of the results  
of the study.  
4. The fourth part is a discussion of the results  
and their implications.  
5. The fifth part is a conclusion and a list of  
references.

6. The sixth part is a list of the authors  
and their affiliations.  
7. The seventh part is a list of the titles  
of the papers presented at the conference.  
8. The eighth part is a list of the titles  
of the papers presented at the workshop.

9. The ninth part is a list of the titles  
of the papers presented at the symposium.  
10. The tenth part is a list of the titles  
of the papers presented at the colloquium.

11. The eleventh part is a list of the titles  
of the papers presented at the round table.  
12. The twelfth part is a list of the titles  
of the papers presented at the panel discussion.





